L Number	Hits	Search Text	DB	Time stamp
-	309584	computer with system	USPAT; US-PGPUB; EPO; JPO; IBM_TDB	2003/08/06 08:44
-	80779	computer with system and computer with network	USPAT; US-PGPUB; EPO; JPO; IBM_TDB	2003/08/06 08:45
-	10631	computer with system and computer with network and sub$system	USPAT; US-PGPUB; EPO; JPO; IBM_TDB	2003/08/06 08:46
-	101	computer with system and computer with network and sub$system and first adj sub$system and second adj sub$system	USPAT; US-PGPUB; EPO; JPO; IBM_TDB	2003/08/06 08:53
-	32	computer with system and computer with network and sub$system and first adj sub$system and second adj sub$system and application adj (software code program)	USPAT; US-PGPUB; EPO; JPO; IBM_TDB	2003/08/06 08:52
-	2	computer with system and computer with network and sub$system and first adj sub$system and second adj sub$system and application adj (software code program) and firewall	USPAT; US-PGPUB; EPO; JPO; IBM_TDB	2003/08/06 08:50
-	2	computer with system and computer with network and sub$system and first adj sub$system and second adj sub$system and application adj (software code program) and fire$wall	USPAT; US-PGPUB; EPO; JPO; IBM_TDB	2003/08/06 08:51
-	97	computer with system and computer with network and sub$system and first adj sub$system and second adj sub$system and (software code program)	USPAT; US-PGPUB; EPO; JPO; IBM_TDB	2003/08/06 08:53
-	473	computer with system and computer with network and sub$system and first near4 sub$system and second near4 sub$system	USPAT; US-PGPUB; EPO; JPO; IBM_TDB	2003/08/06 08:54
-	468	computer with system and computer with network and sub$system and first near4 sub$system and second near4 sub$system and (application software program code instruction)	USPAT; US-PGPUB; EPO; JPO; IBM_TDB	2003/08/06 08:56
-	14	computer with system and computer with network and sub$system and first near4 sub$system and second near4 sub$system and (application software program code instruction) and firewall	USPAT; US-PGPUB; EPO; JPO; IBM_TDB	2003/08/06 08:56
-	9	computer with system and computer with network and sub$system and first near4 sub$system and second near4 sub$system and (application software program code instruction) and firewall and (http "hypertext transfer protocol")	USPAT; US-PGPUB; EPO; JPO; IBM_TDB	2003/08/06 09:09
-	4	computer with system and computer with network and sub$system with ((application software) with code instruction) and subsystem with ((output result display) and firewall) and (http "hypertext transfer protocol")	USPAT; US-PGPUB; EPO; JPO; IBM_TDB	2003/08/06 09:13
-	59	computer with system and computer with network and sub$system with (application software) AND (code instruction) and subsystem with (output result display) and firewall and (http "hypertext transfer protocol")	USPAT; US-PGPUB; EPO; JPO; IBM_TDB	2003/08/06 09:15

-	4	computer with system and computer with network and sub$system with (application software) AND (code instruction) and subsystem with (output result display) and firewall and (http "hypertext transfer protocol") AND SOCKET and "http 1.1"	USPAT; US-PGPUB; EPO; JPO; IBM_TDB	2003/08/06 09:16
-	39	computer with system and computer with network and sub$system with (application software) AND (code instruction) and subsystem with (output result display) and firewall and (http "hypertext transfer protocol") AND SOCKET	USPAT; US-PGPUB; EPO; JPO; IBM_TDB	2003/08/06 09:49
-	5	713/$.ccls. and network and sub$system with (application software) AND (code instruction) and subsystem with (output result display) and firewall and (http "hypertext transfer protocol") AND SOCKET	USPAT; US-PGPUB; EPO; JPO; IBM_TDB	2003/08/06 09:53
-	5	713/$.ccls. and network and sub$system with (application software code instruction) and subsystem with (output result display) and firewall and (http "hypertext transfer protocol") AND SOCKET	USPAT; US-PGPUB; EPO; JPO; IBM_TDB	2003/08/06 09:54
-	7	713/$.ccls. and network and sub$system with (application software code instruction) and subsystem with (output result display) and firewall and (http "hypertext transfer protocol")	USPAT; US-PGPUB; EPO; JPO; IBM_TDB	2003/08/06 12:28
-	102	network and sub$system with (application software code instruction) and subsystem with (output result display) and firewall and (http "hypertext transfer protocol")	USPAT; US-PGPUB; EPO; JPO; IBM_TDB	2003/08/06 10:04
-	9	network and sub$system with (application software code instruction) and subsystem with (output result display) and firewall and (http "hypertext transfer protocol") and socket and thread	USPAT; US-PGPUB; EPO; JPO; IBM_TDB	2003/08/06 09:55
-	6	network and sub$system with (application software code instruction) and subsystem with (output result display) and firewall and (http "hypertext transfer protocol") and "http 1.1"	USPAT; US-PGPUB; EPO; JPO; IBM_TDB	2003/08/06 10:01
-	48	network and sub$system with (application software code instruction) and subsystem with (output result display) and firewall and (http "hypertext transfer protocol") and socket	USPAT; US-PGPUB; EPO; JPO; IBM_TDB	2003/08/06 10:04
-	1	5950173.pn.	USPAT; US-PGPUB; EPO; JPO; IBM_TDB	2003/08/06 10:48
-	1	713/$.ccls. and network and sub$system with (application software code instruction) and subsystem with (output result display) and firewall and (http "hypertext transfer protocol") and state with communicat$4	USPAT; US-PGPUB; EPO; JPO; IBM_TDB	2003/08/06 12:43
-	7	state$ful near communicat$4	USPAT; US-PGPUB; EPO; JPO; IBM_TDB	2003/08/06 12:58
-	3955	"port 80"	USPAT; US-PGPUB; EPO; JPO; IBM_TDB	2003/08/06 12:59
-	448	"port 80" and http	USPAT; US-PGPUB; EPO; JPO; IBM_TDB	2003/08/06 12:59

-	271	"port 80" and http and state	USPAT; US-PGPUB; EPO; JPO; IBM_TDB	2003/08/06 12:59
-	18	"port 80" and http and state$ful	USPAT; US-PGPUB; EPO; JPO; IBM_TDB	2003/08/06 13:15
-	3	"port 80" near4 firewall and http and state$ful	USPAT; US-PGPUB; EPO; JPO; IBM_TDB	2003/08/06 13:09
-	0	"port 80" near4 firewall and "http 1.1" and state$ful	USPAT; US-PGPUB; EPO; JPO; IBM_TDB	2003/08/06 13:15
-	0	"port 80" with firewall and http near "1.1" and state$ful	USPAT; US-PGPUB; EPO; JPO; IBM_TDB	2003/08/06 13:16
-	3	"port 80" with firewall and http and state$ful	USPAT; US-PGPUB; EPO; JPO; IBM_TDB	2003/08/06 13:17
-	3	"port 80" with firewall and state$ful	USPAT; US-PGPUB; EPO; JPO; IBM_TDB	2003/08/06 13:17
-	5	"port 80" with firewall and http with "1.1"	USPAT; US-PGPUB; EPO; JPO; IBM_TDB	2003/08/06 13:17

US 20020108059A1

(19) **United States**

(12) **Patent Application Publication** (10) **Pub. No.: US 2002/0108059 A1**

Canion et al. (43) **Pub. Date: Aug. 8, 2002**

(54) **NETWORK SECURITY ACCELERATOR**

(76) Inventors: **Rodney S. Canion**, West Lake Hills, TX (US); **Brian W. Bailey**, Austin, TX (US); **Thomas E. Garvens**, Austin, TX (US); **Scott C. Johnson**, Round Rock, TX (US); **Roger K. Richter**, Leander, TX (US); **Ho Wang**, Austin, TX (US)

Correspondence Address:
Richard D. Egan
O'KEEFE, EGAN, PETERMAN
1101 Capital of Texas Highway South
Building C, Suite 200
Austin, TX 78746 (US)

(21) Appl. No.: **09/797,411**

(22) Filed: **Mar. 1, 2001**

Related U.S. Application Data

(60) Provisional application No. 60/246,335, filed on Nov. 7, 2000. Provisional application No. 60/187,211, filed on Mar. 3, 2000.

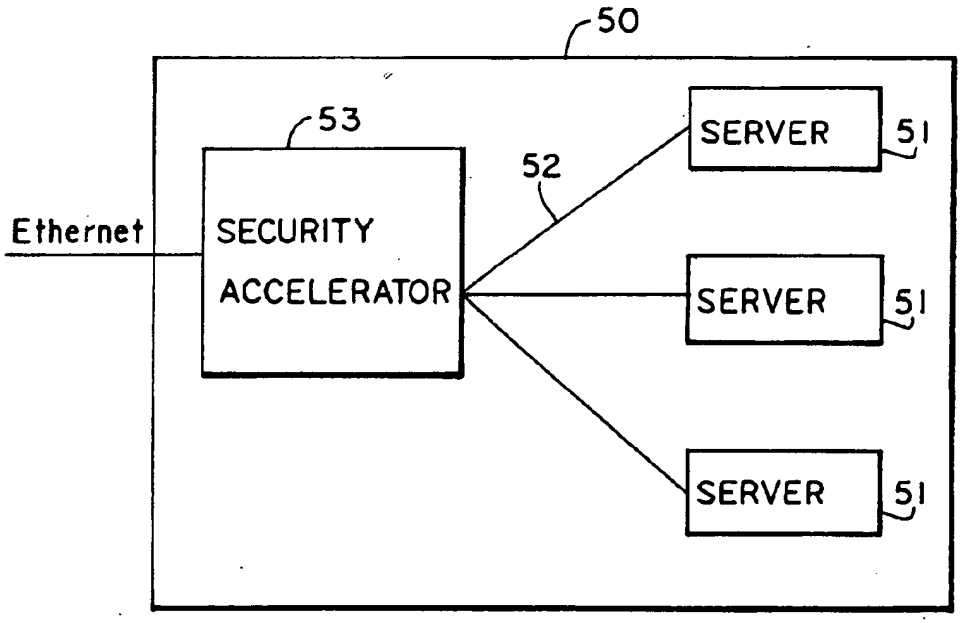
Publication Classification

(51) **Int. Cl.**[7] **G06F 11/30**

(52) **U.S. Cl.** **713/201**; 709/224

(57) **ABSTRACT**

A network processing system uses intelligent security hardware as a security accelerator at its front end. The security hardware performs initial processing of incoming data, such as security detection tasks. The security hardware is directly connected to one or more processing units, via a bus or switch fabric, which execute appropriate applications and/or storage programming.



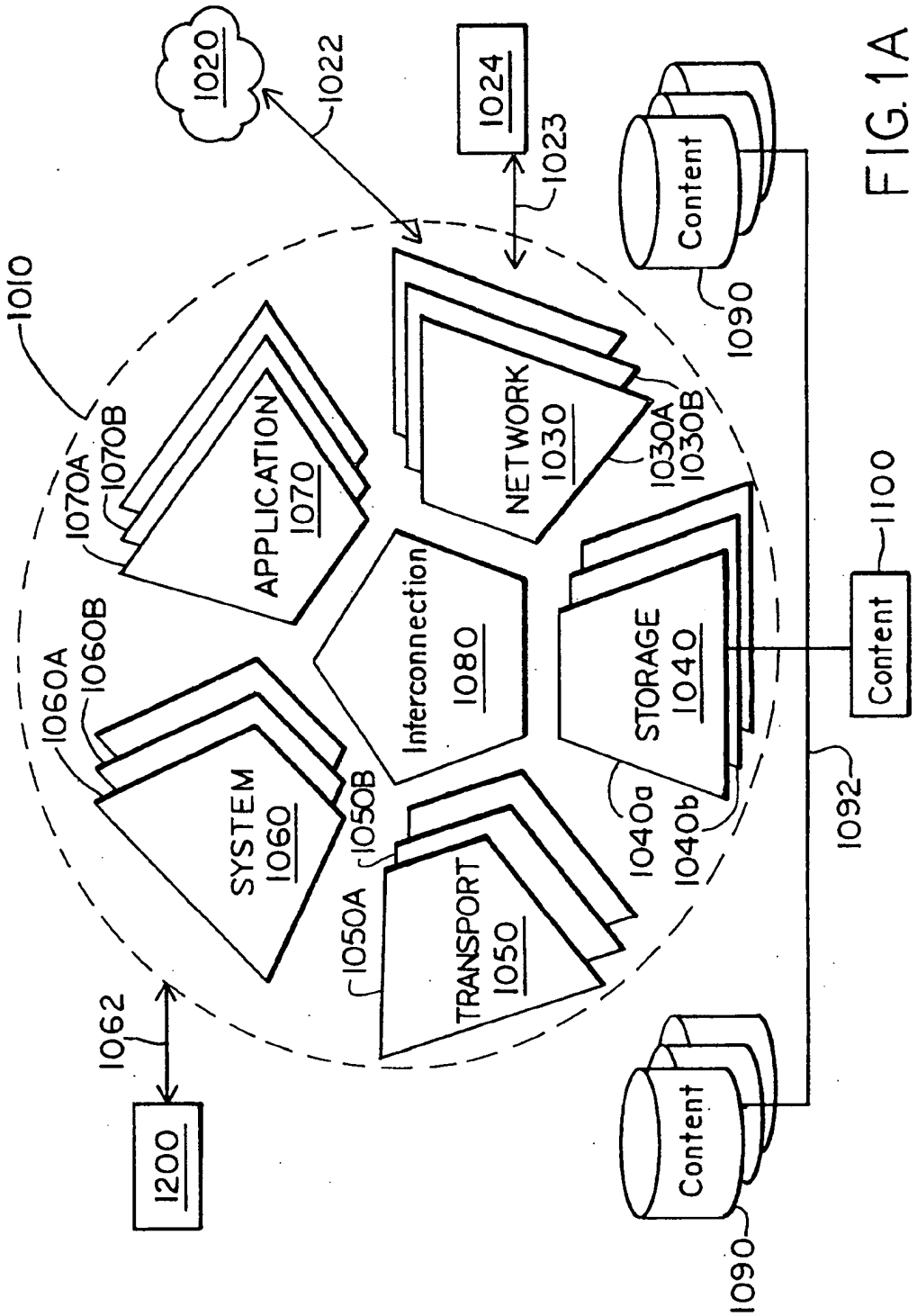

1020
1022
1024
1023
Content
1090
1010
1070A
1070B
APPLICATION
1070
NETWORK
1030
1030A
1030B
1100
Content
1060A
1060B
SYSTEM
1060
Interconnection
1080
STORAGE
1040
1040a
1040b
1092
1050B
1050A
TRANSPORT
1050
1062
1200
Content
1090

FIG. 1A

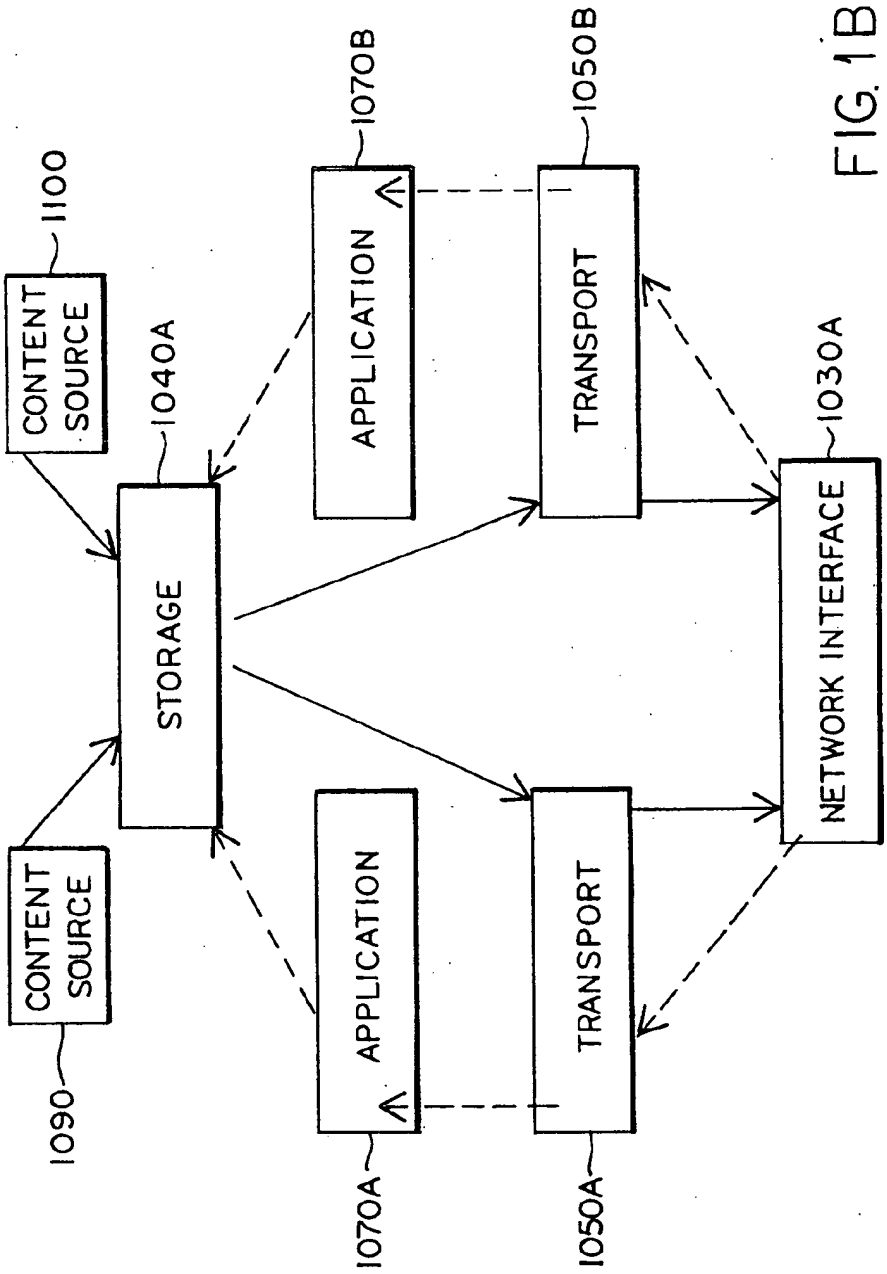

1100
CONTENT SOURCE
1040A
STORAGE
1070B
APPLICATION
1050B
TRANSPORT
1030A
NETWORK INTERFACE
CONTENT SOURCE
1090
APPLICATION
1070A
TRANSPORT
1050A

FIG. 1B

FIG.1C | FIG. 1C' | FIG. 1C''

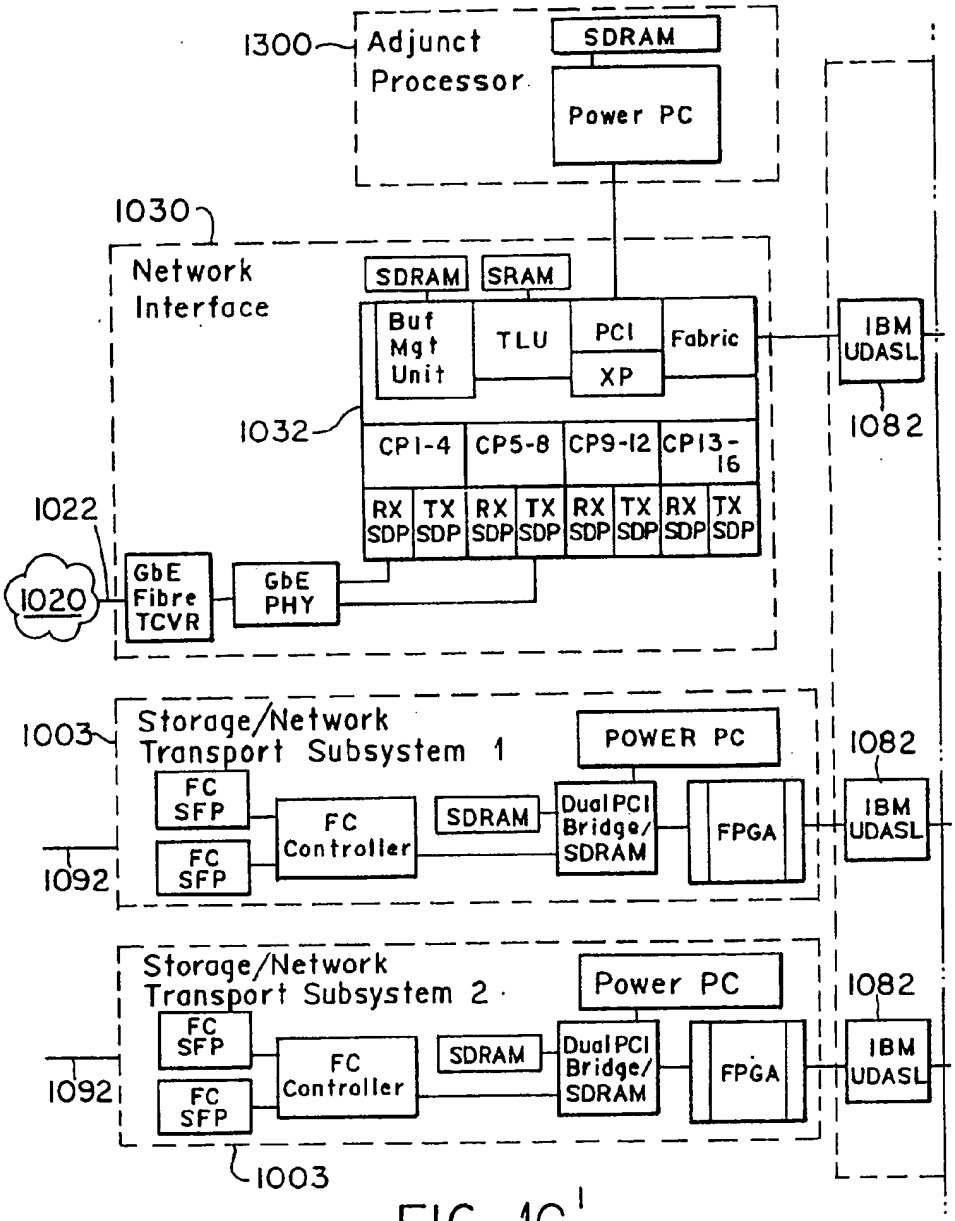


FIG. 1C'

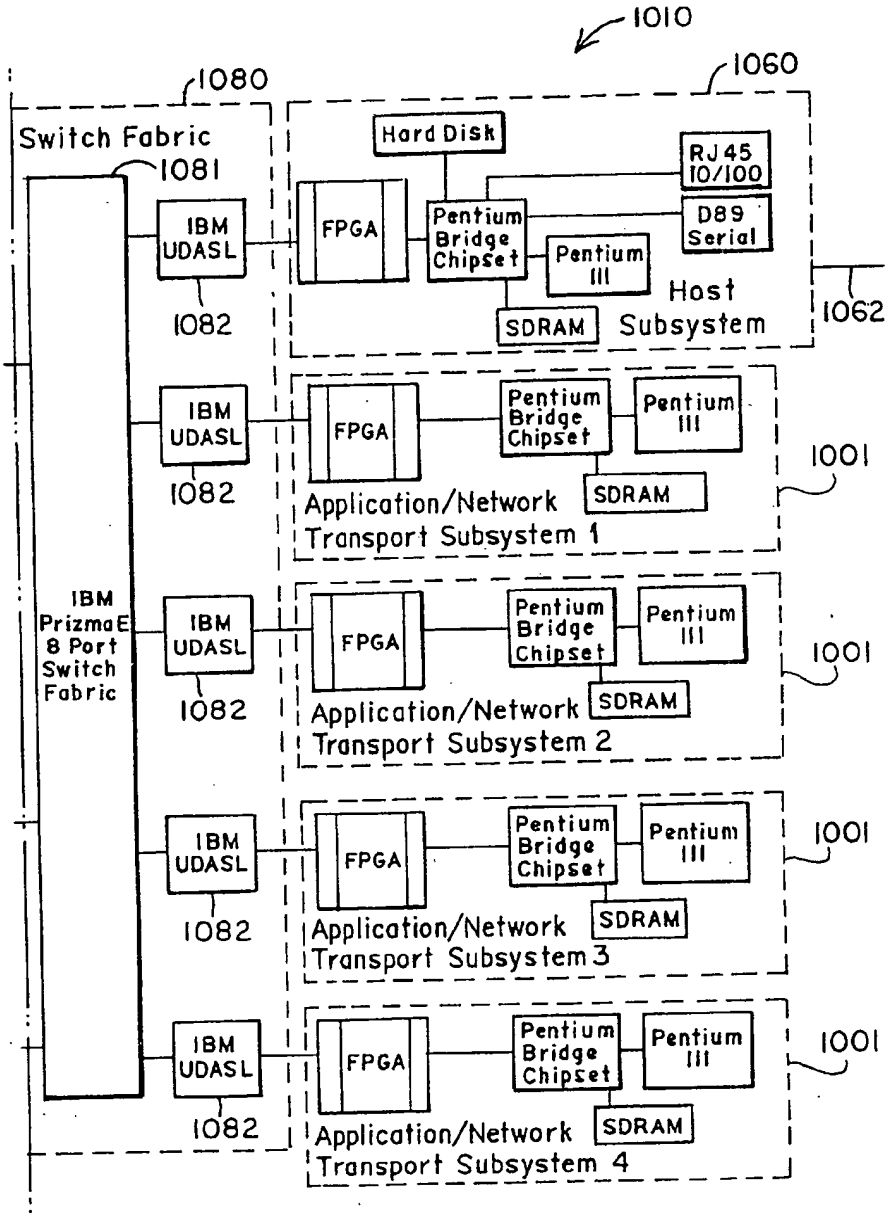

1010
1080
1060
Switch Fabric
1081
Hard Disk
RJ45 10/100
D89 Serial
IBM UDASL
FPGA
Pentium Bridge Chipset
Pentium III
SDRAM
Host Subsystem
1062
1082
Application/Network Transport Subsystem 1
1001
IBM PrizmaE 8 Port Switch Fabric
Application/Network Transport Subsystem 2
Application/Network Transport Subsystem 3
Application/Network Transport Subsystem 4

FIG. 1C II

FIG.1D | FIG.1D' | FIG.1D''

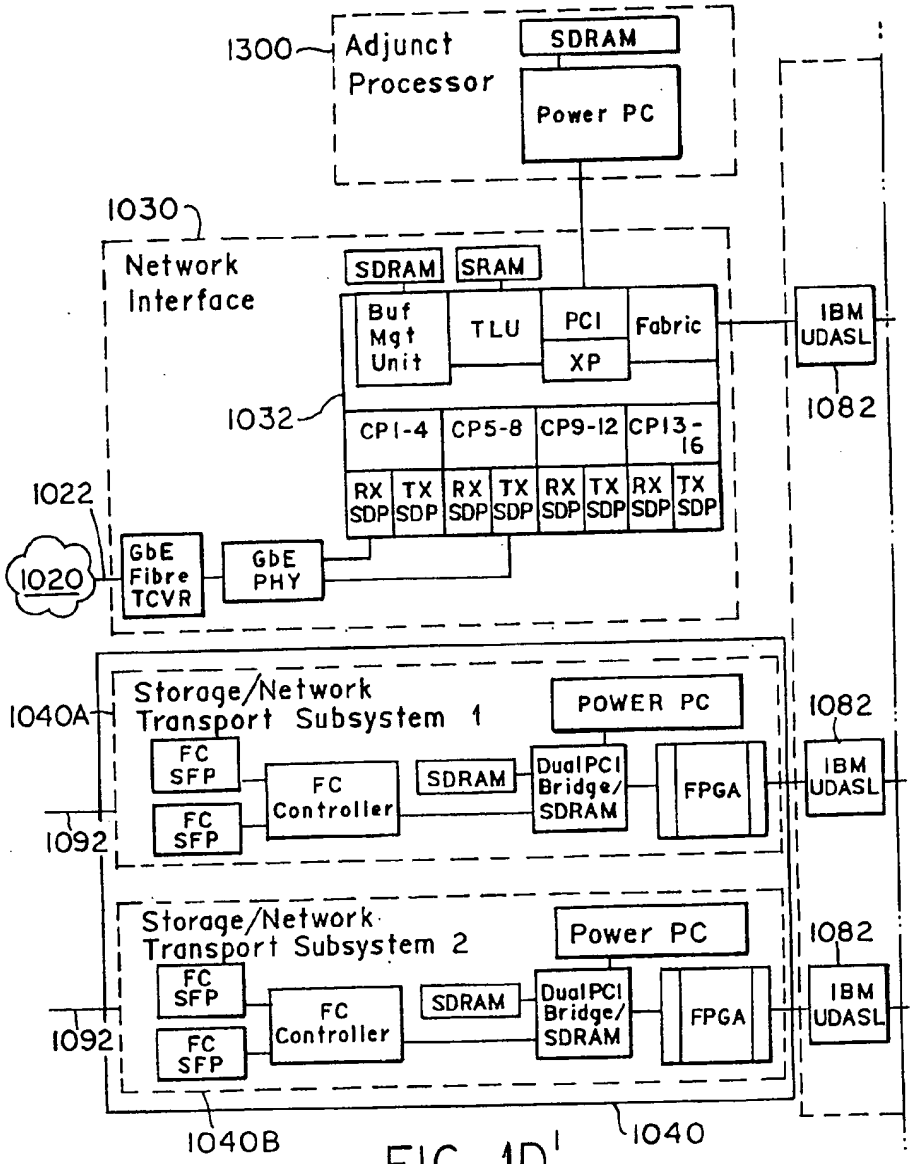


FIG. 1D'

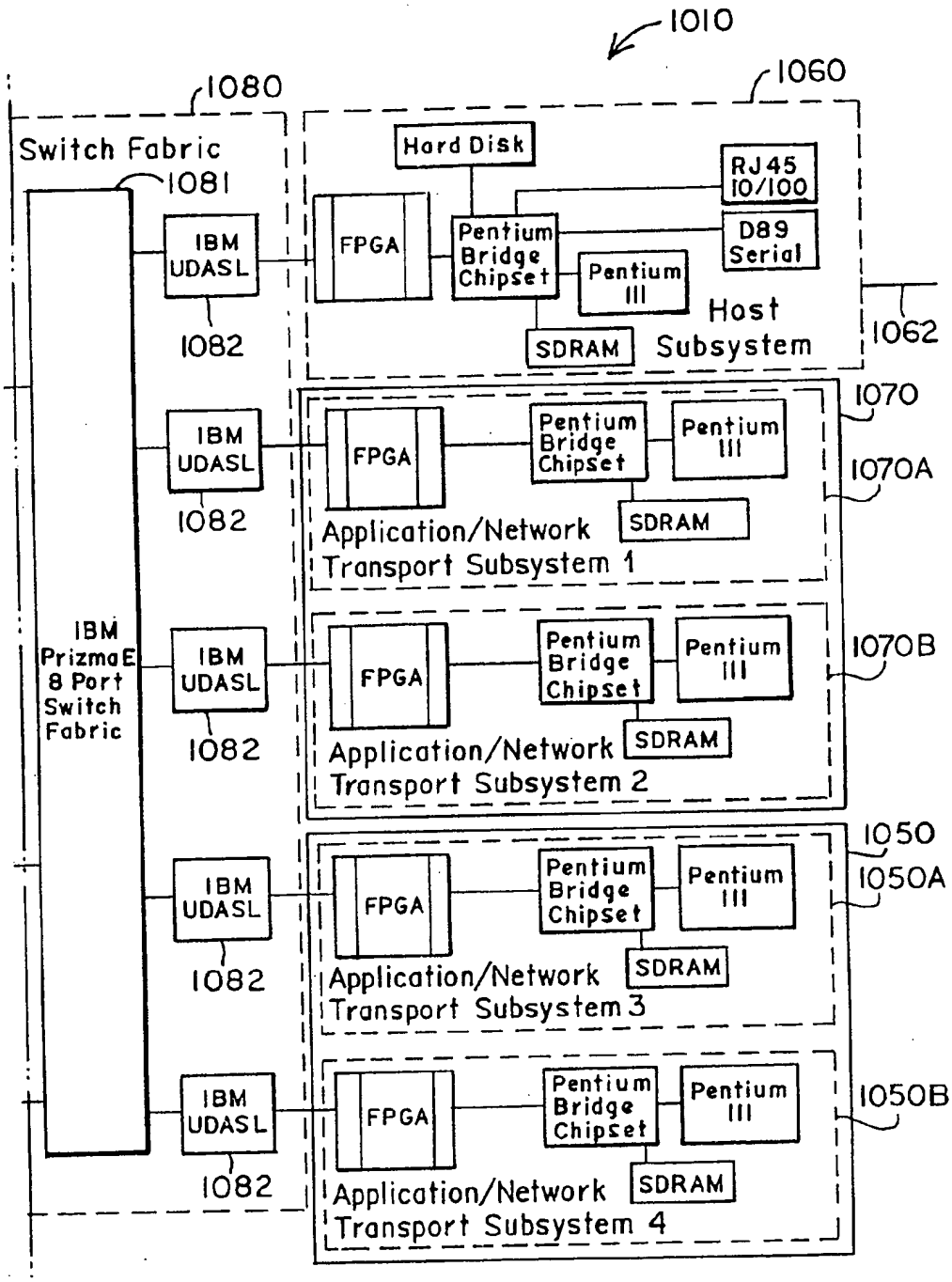

1010
1080
1060
Switch Fabric
1081
Hard Disk
RJ45 10/100
D89 Serial
FPGA
Pentium Bridge Chipset
Pentium III
SDRAM
Host Subsystem
1062
IBM UDASL
1082
1070
1070A
Application/Network Transport Subsystem 1
1070B
Application/Network Transport Subsystem 2
IBM PrizmaE 8 Port Switch Fabric
1050
1050A
Application/Network Transport Subsystem 3
1050B
Application/Network Transport Subsystem 4

FIG. 1D[11]

FIG.1E | FIG. 1E' | FIG. 1E''

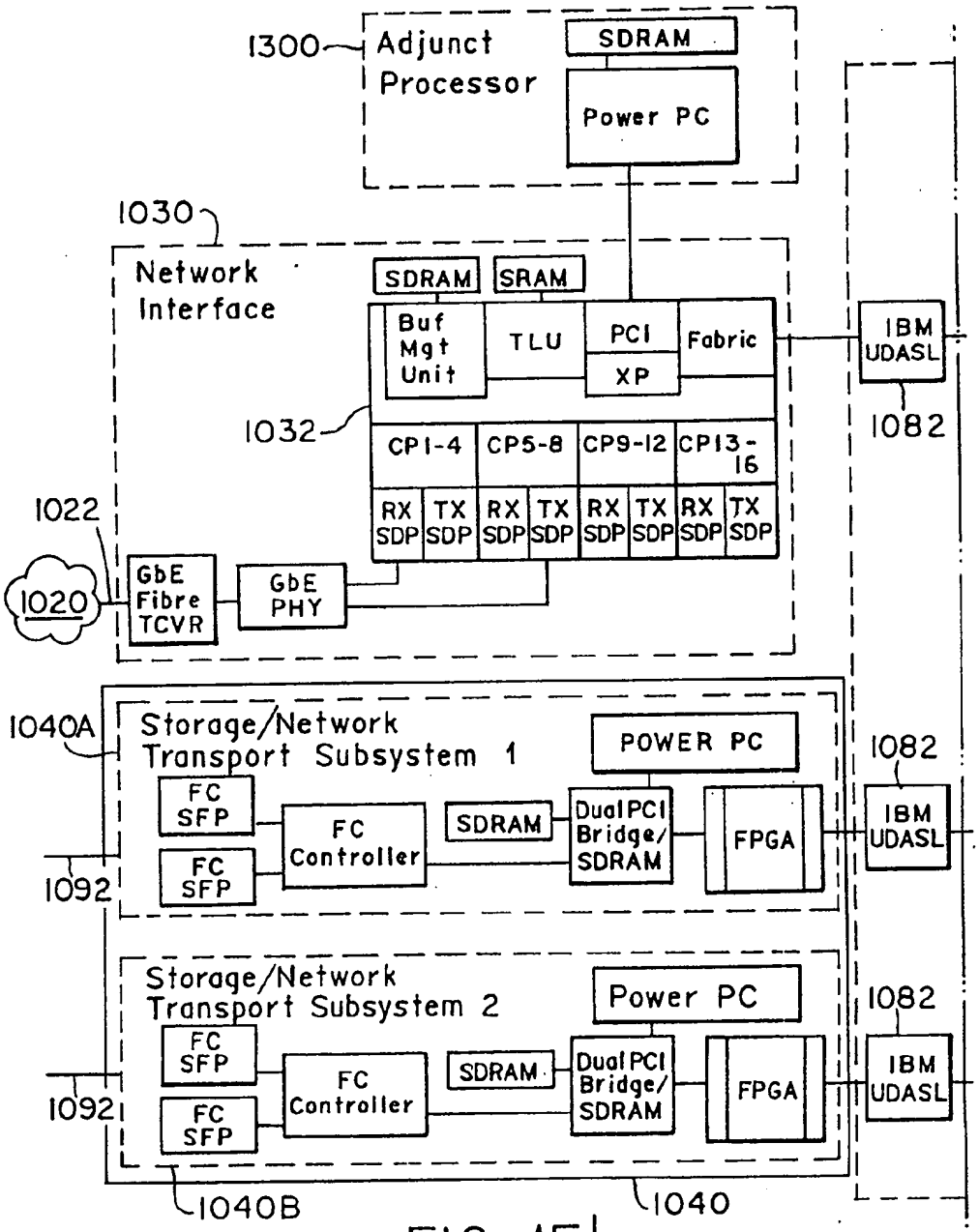


FIG. 1E'

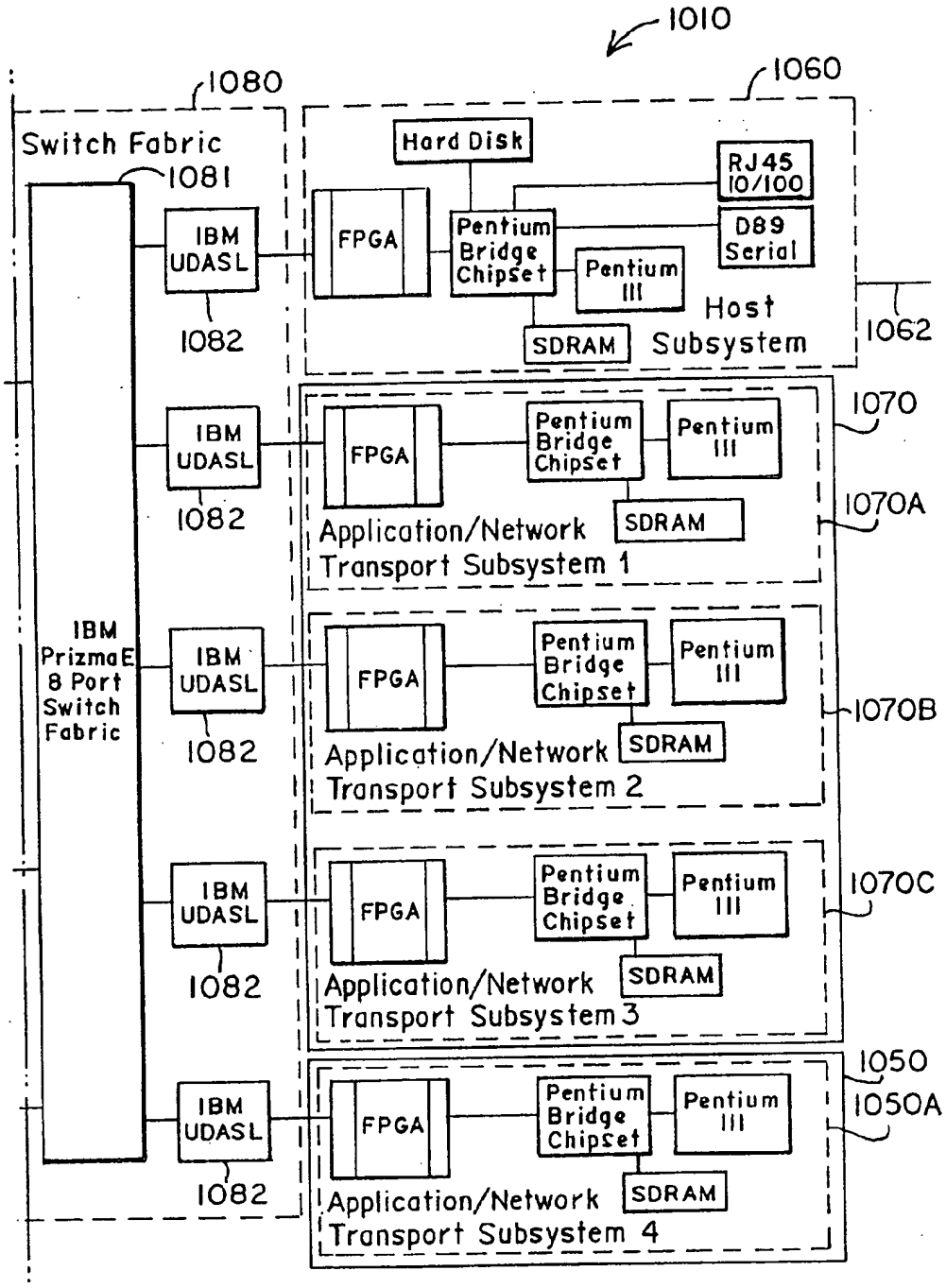

1010
1080
1060
Switch Fabric
1081
Hard Disk
RJ45 10/100
D89 Serial
FPGA
Pentium Bridge Chipset
Pentium III
SDRAM
Host Subsystem
1062
IBM UDASL
1082
1070
1070A
Application/Network Transport Subsystem 1
1070B
Application/Network Transport Subsystem 2
1070C
Application/Network Transport Subsystem 3
1050
1050A
Application/Network Transport Subsystem 4
IBM PrizmaE 8 Port Switch Fabric

FIG. 1E[11]

FIG.1F | FIG.1F' | FIG.1F''

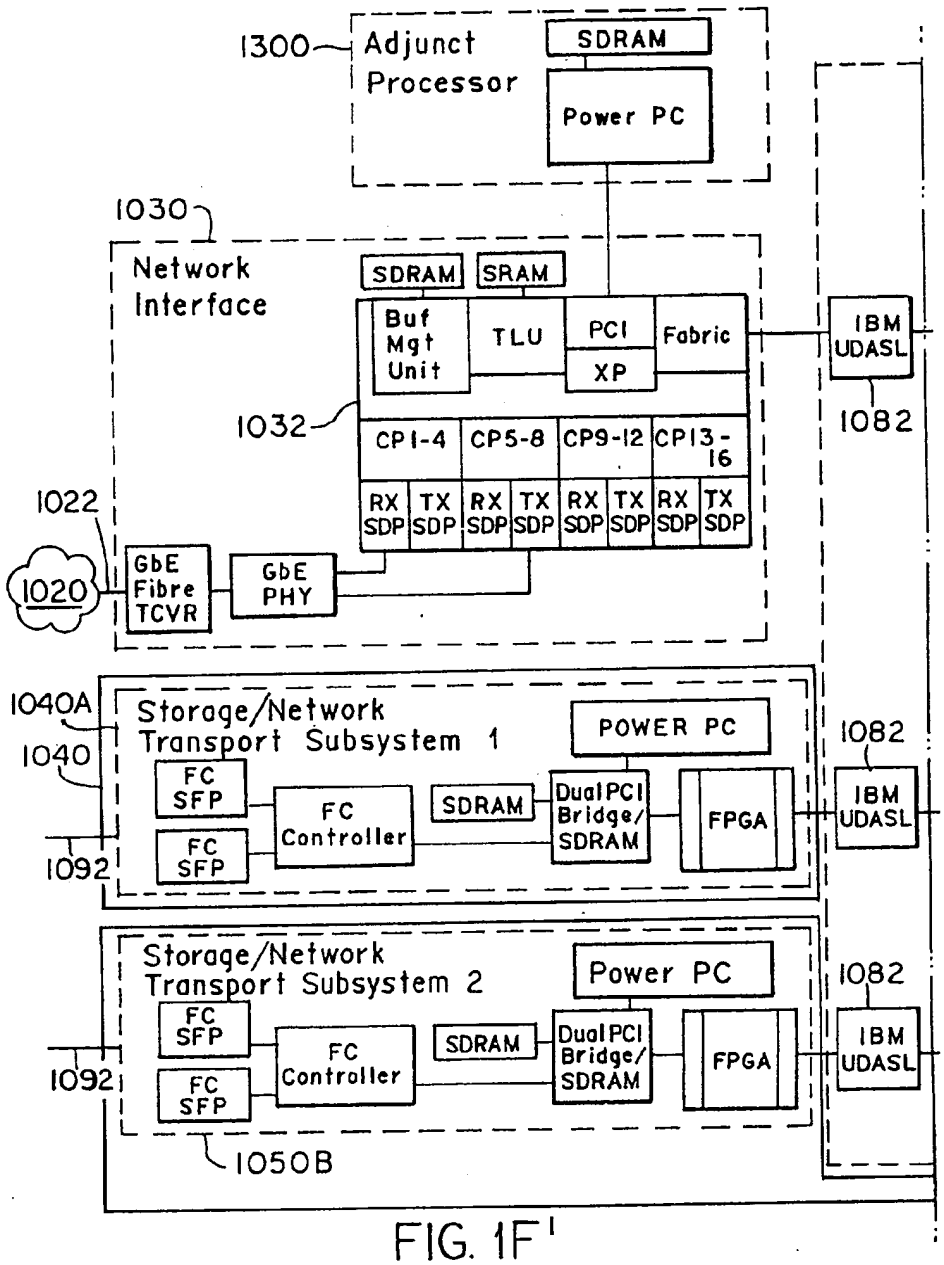


FIG. 1F'

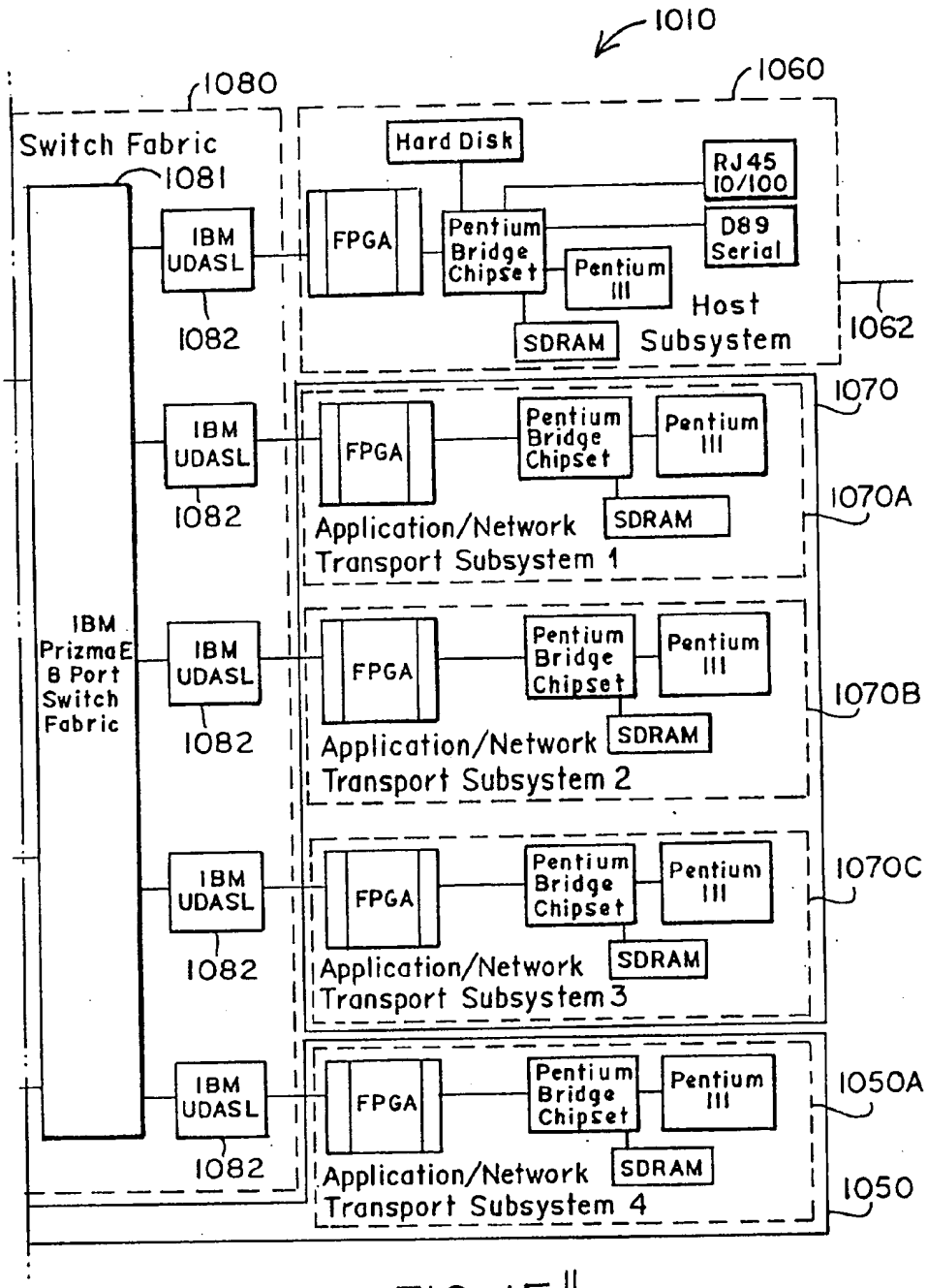

1010
1080
1060
Switch Fabric
1081
Hard Disk
RJ45 10/100
D89 Serial
IBM UDASL
FPGA
Pentium Bridge Chipset
Pentium III
SDRAM
Host Subsystem
1062
1082
1070
1070A
Application/Network Transport Subsystem 1
1070B
Application/Network Transport Subsystem 2
1070C
Application/Network Transport Subsystem 3
1050A
Application/Network Transport Subsystem 4
1050
IBM PrizmaE 8 Port Switch Fabric

FIG. 1F II

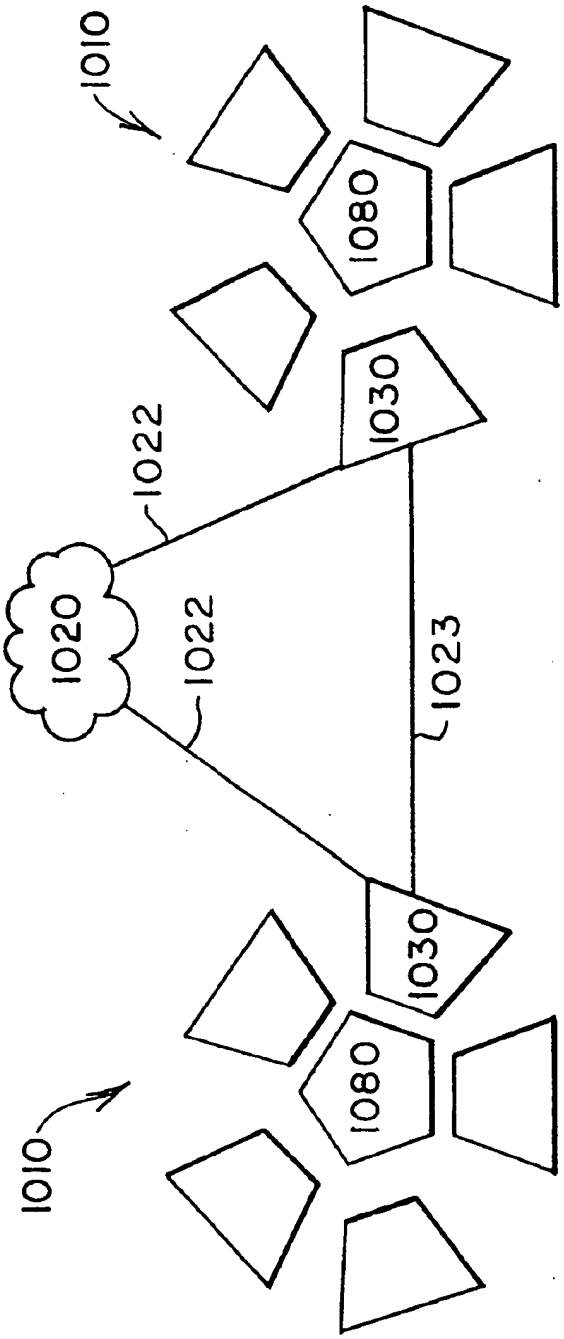


FIG. 1G

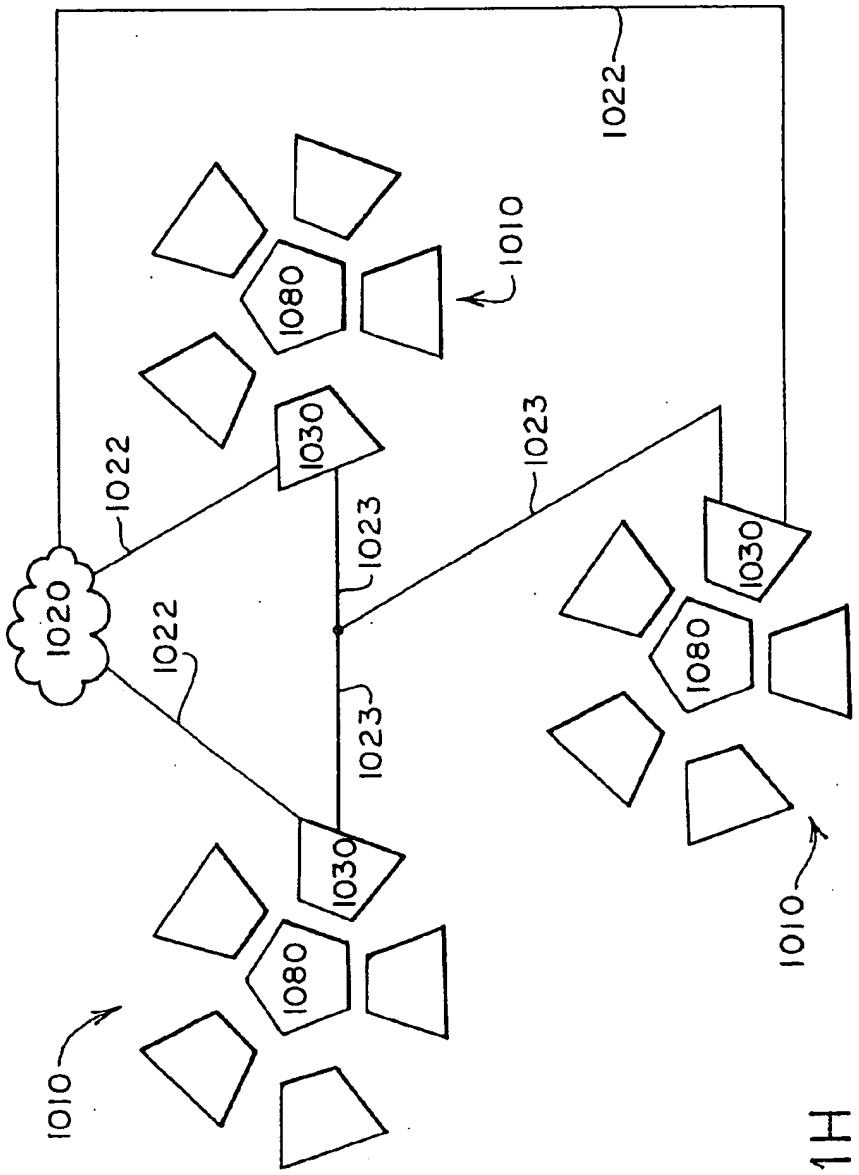

1022
1010
1080
1030
1023
1022
1020
1022
1023
1030
1080
1030
1080
1010
1010

FIG. 1H

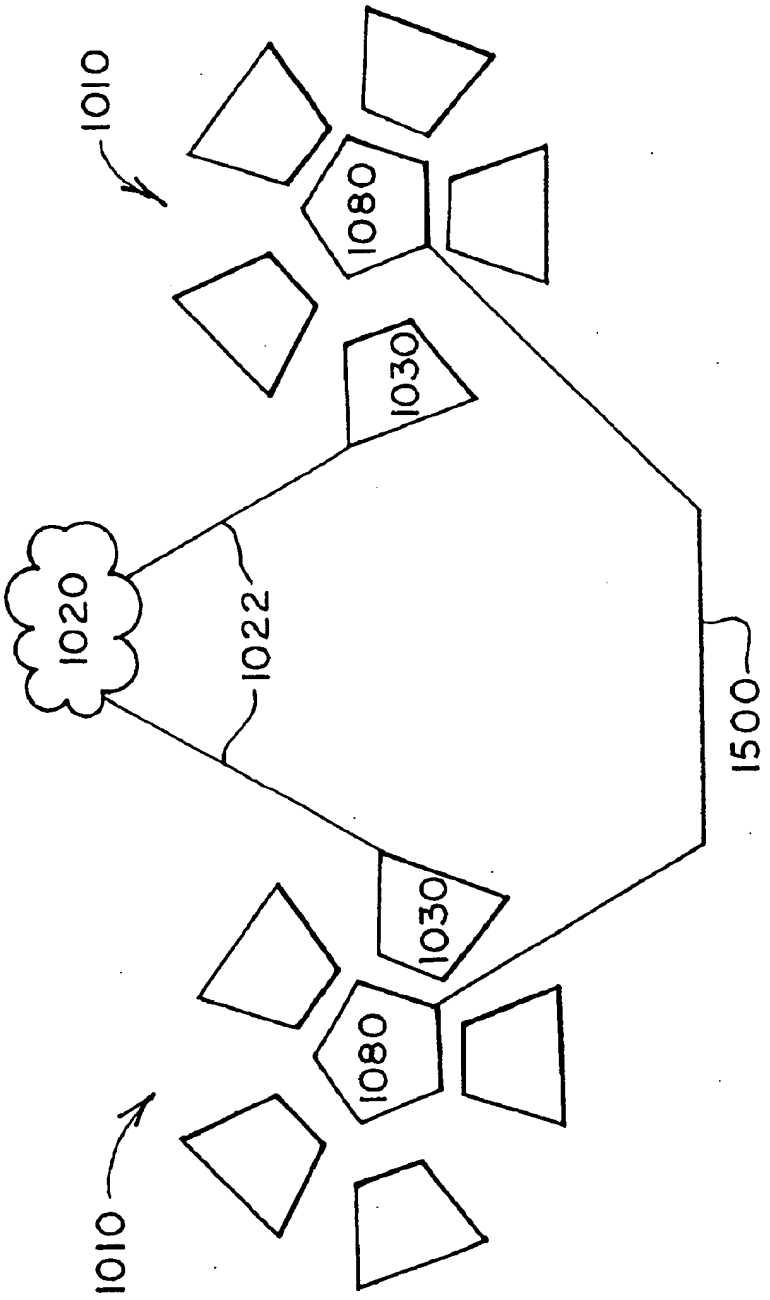


FIG. 11

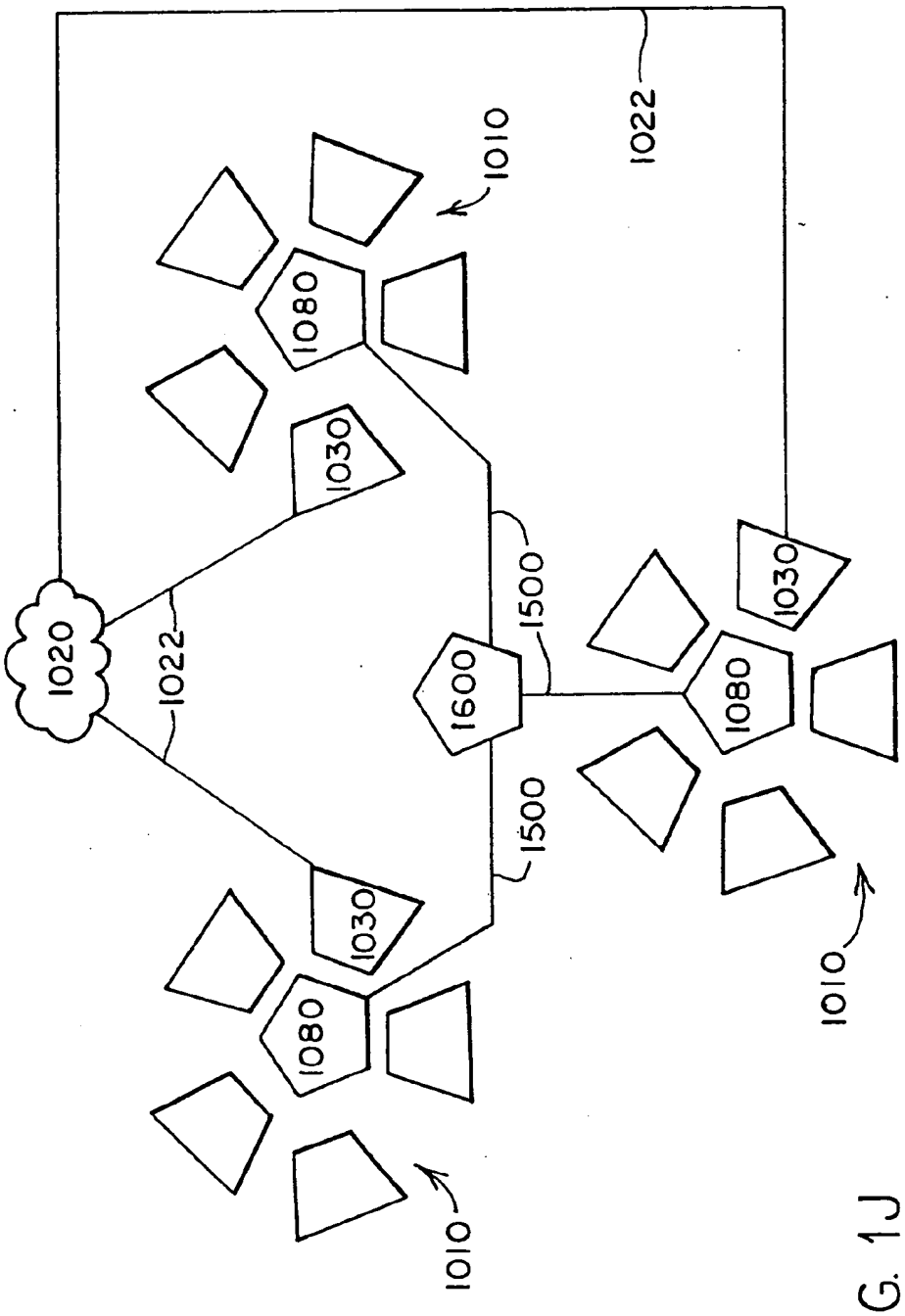

1022
1010
1080
1030
1020
1022
1500
1600
1030
1080
1500
1030
1080
1010
1010

FIG. 1J

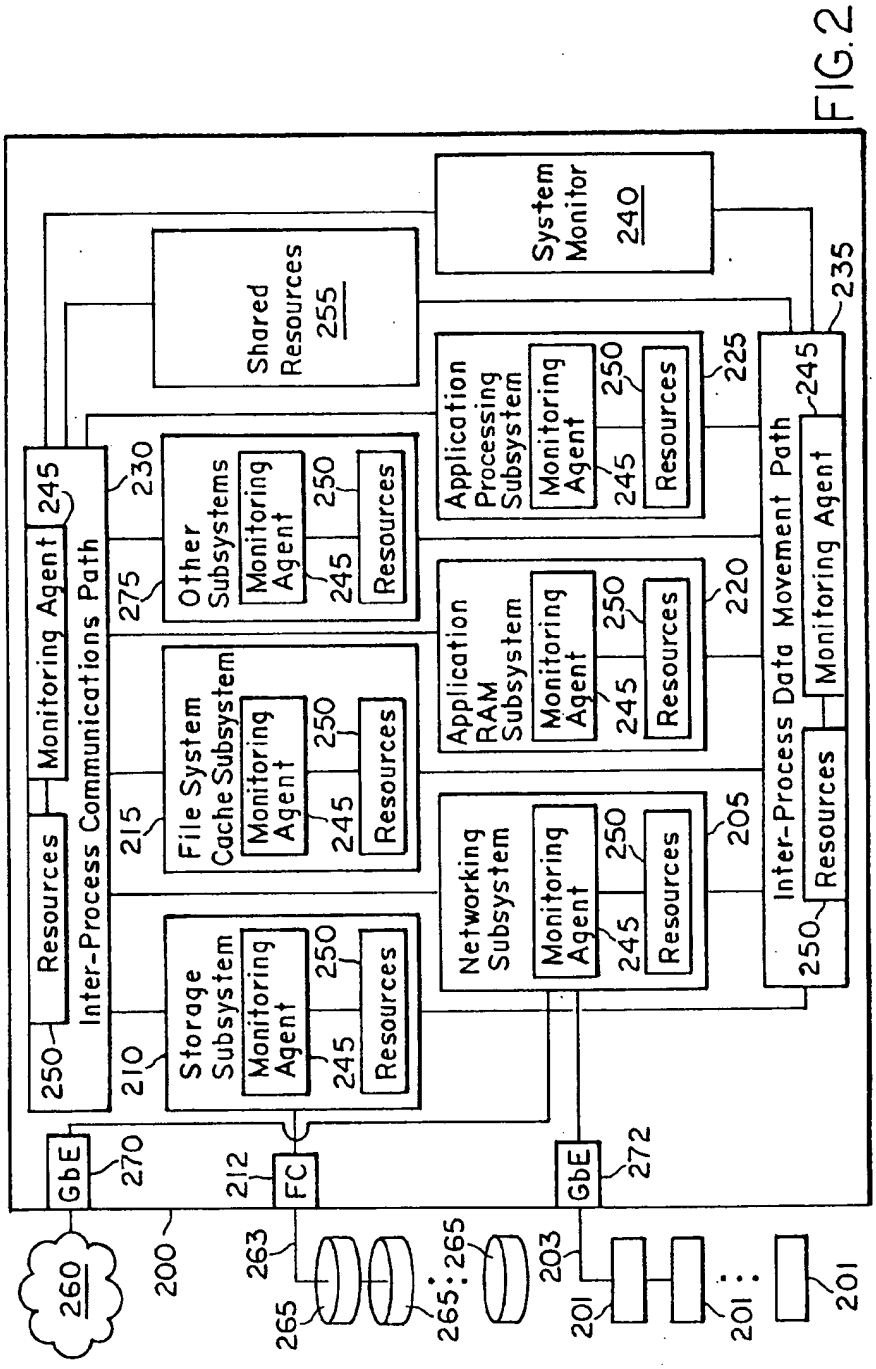

System Monitor 240
Shared Resources 255
Application Processing Subsystem
Monitoring Agent
Resources
Other Subsystems
Application RAM Subsystem
File System Cache Subsystem
Networking Subsystem
Storage Subsystem
Inter-Process Communications Path
Inter-Process Data Movement Path
GbE
FC
260
200
263
265
203
201
235
230
225
220
215
205
210
270
272
212

FIG.2

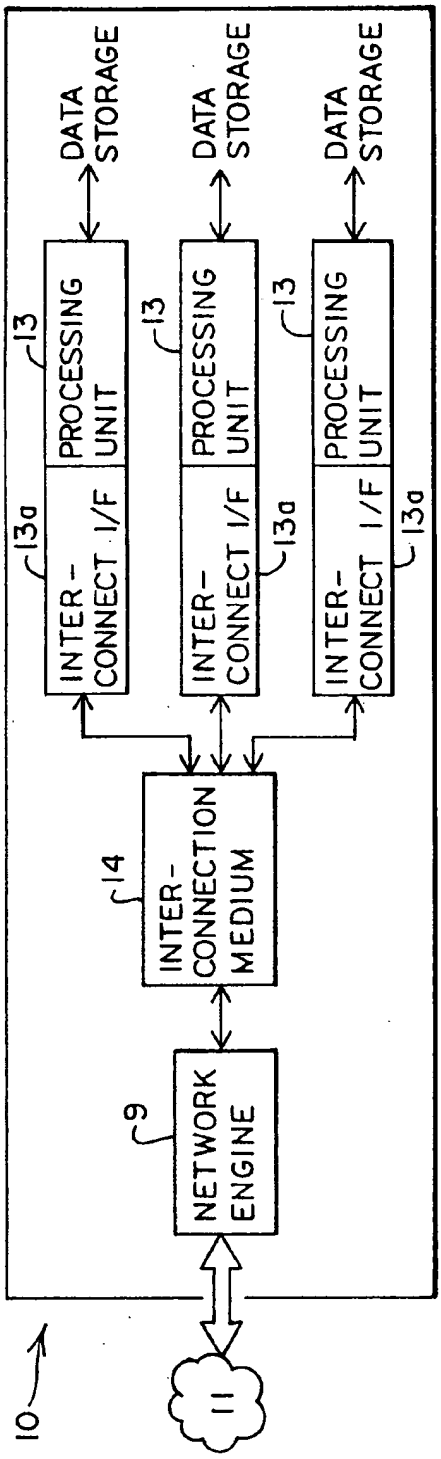

10
11
9
NETWORK ENGINE
14
INTER-CONNECTION MEDIUM
13a
INTER-CONNECT I/F
13
PROCESSING UNIT
DATA STORAGE

FIG. 2A

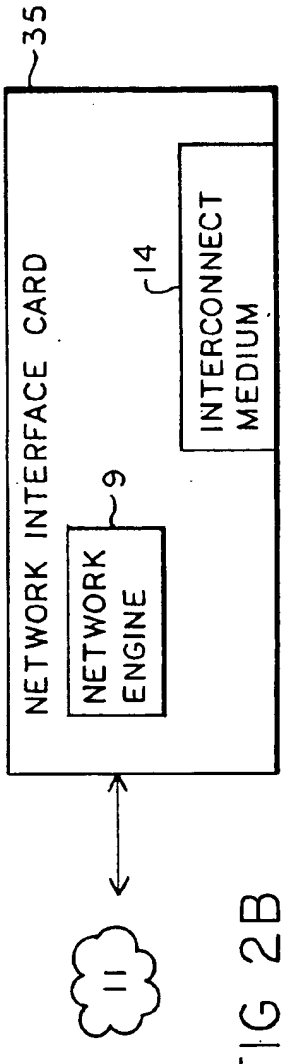

35
NETWORK INTERFACE CARD
9
NETWORK ENGINE
14
INTERCONNECT MEDIUM
11

FIG 2B

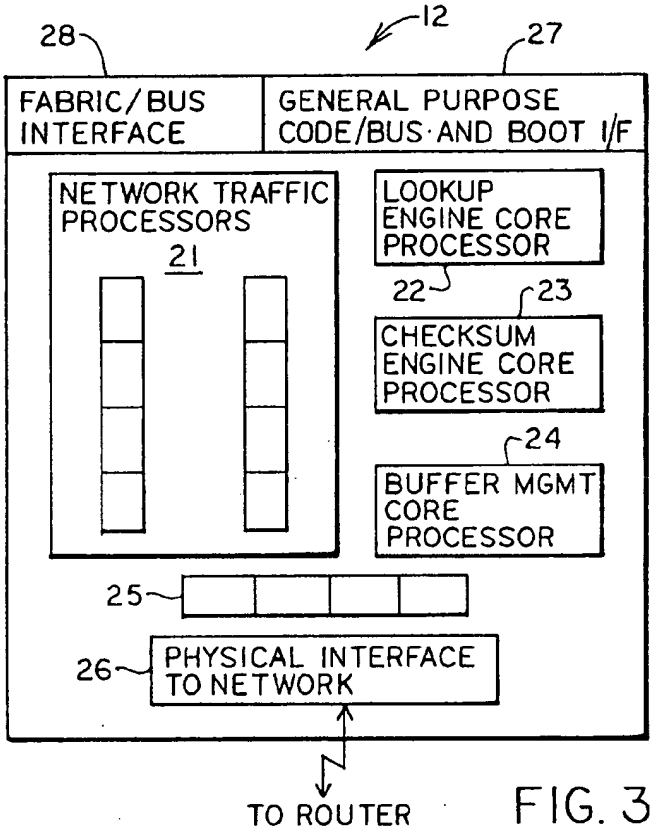

12
28
27
FABRIC/BUS INTERFACE
GENERAL PURPOSE CODE/BUS AND BOOT I/F
NETWORK TRAFFIC PROCESSORS
21
LOOKUP ENGINE CORE PROCESSOR
22
23
CHECKSUM ENGINE CORE PROCESSOR
24
BUFFER MGMT CORE PROCESSOR
25
26
PHYSICAL INTERFACE TO NETWORK
TO ROUTER

FIG. 3

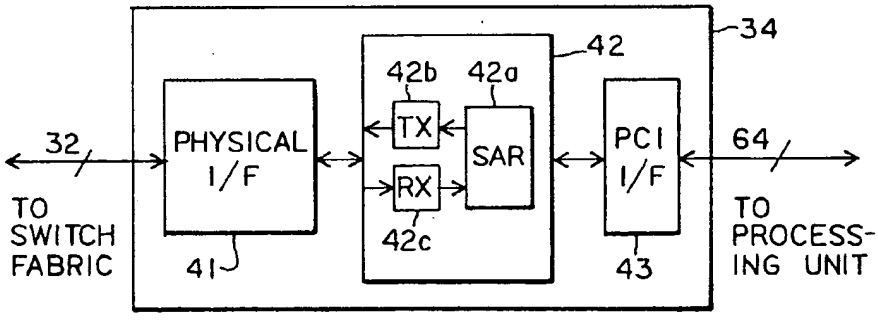

34
42
42b
42a
TX
RX
SAR
32
PHYSICAL I/F
PCI I/F
64
TO SWITCH FABRIC
TO PROCESSING UNIT
41
42c
43

FIG. 4

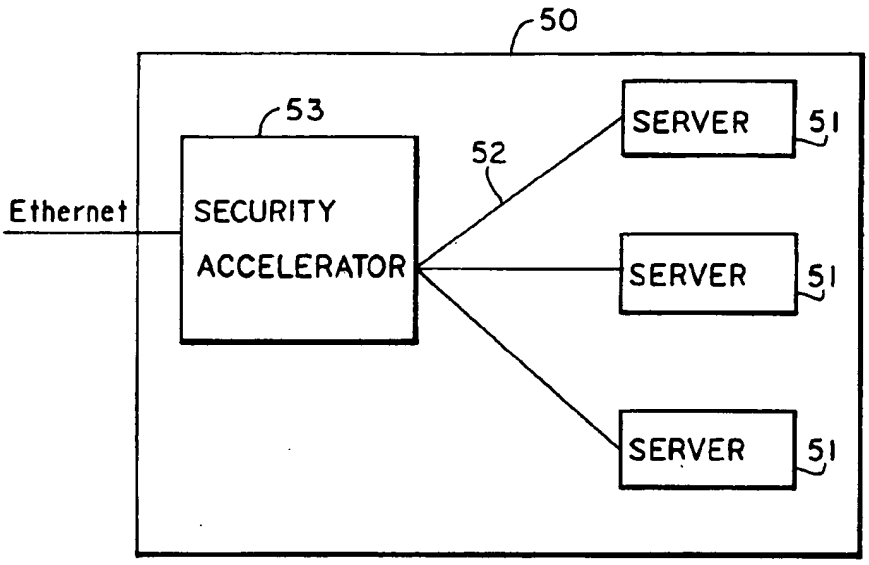

50
53
52
Ethernet
SECURITY
ACCELERATOR
SERVER
51
SERVER
51
SERVER
51

FIG. 5

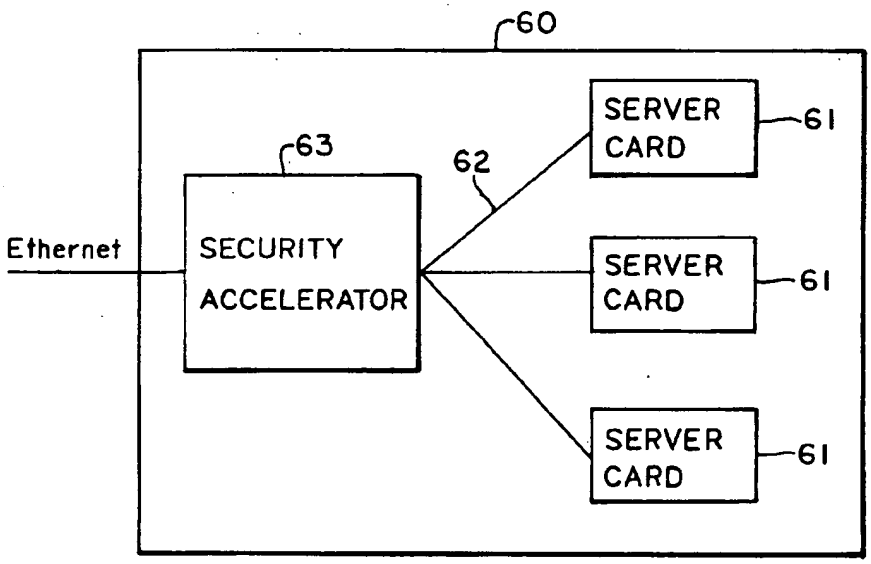

60
63
62
Ethernet
SECURITY
ACCELERATOR
SERVER
CARD
61
SERVER
CARD
61
SERVER
CARD
61

FIG. 6

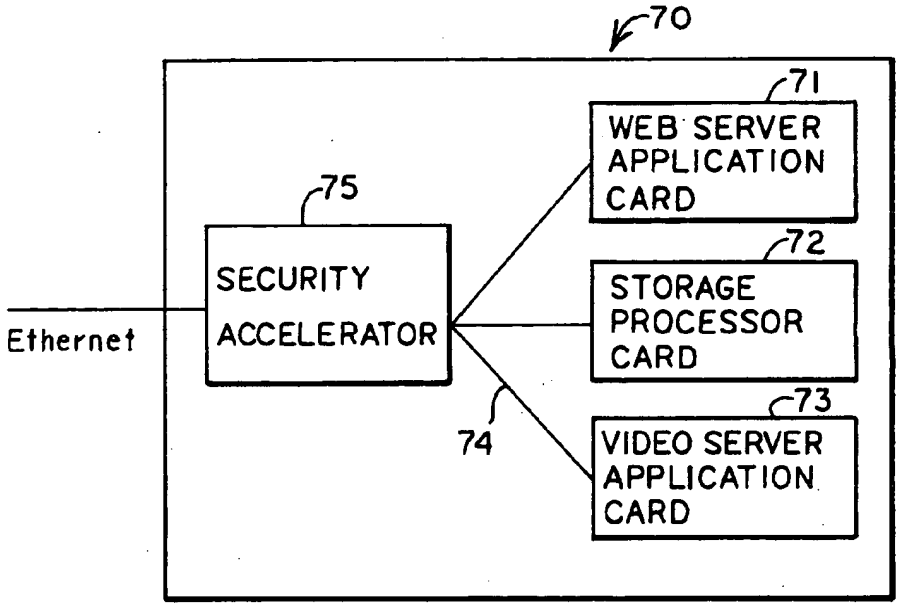

70
71
WEB SERVER APPLICATION CARD
75
SECURITY ACCELERATOR
72
STORAGE PROCESSOR CARD
Ethernet
74
73
VIDEO SERVER APPLICATION CARD

FIG. 7

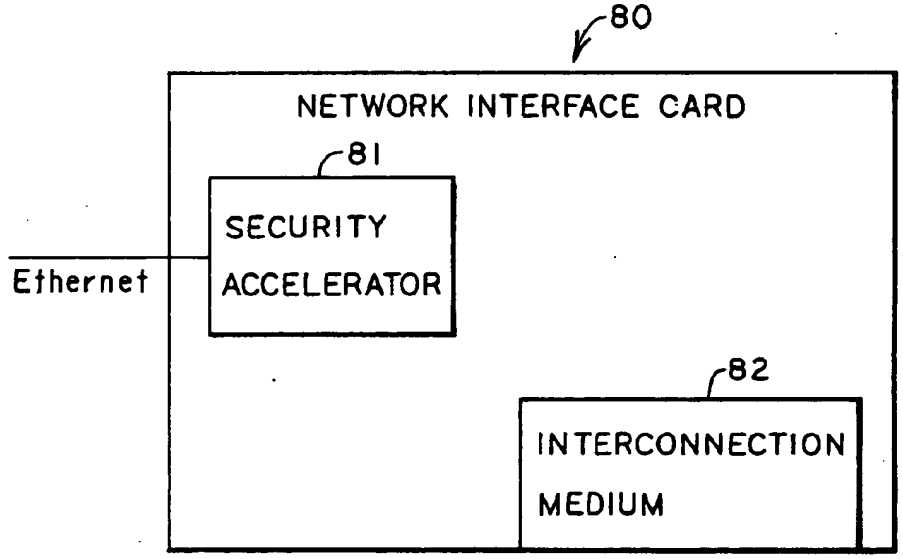

80
NETWORK INTERFACE CARD
81
SECURITY ACCELERATOR
Ethernet
82
INTERCONNECTION MEDIUM

FIG. 8

NETWORK SECURITY ACCELERATOR

[0001] This application claims priority from Provisional Application Serial No. 60/246,335 filed on Nov. 7, 2000 which is entitled "NETWORK SECURITY ACCELERATOR" and to Provisional Application Serial No. 60/187,211 filed on Mar. 3, 2000 which is entitled "SYSTEM AND APPARATUS FOR INCREASING FILE SERVER BANDWIDTH," the disclosures of each being incorporated herein by reference.

BACKGROUND OF THE INVENTION

[0002] The present invention relates generally to network connected computing systems, and more particularly, to security techniques for network connected computing systems.

[0003] Network connected computing systems often incorporate a variety of security measures in order to protect against a wide variety of unauthorized intrusions, access, or attacks against the computing system. An important task of a network connected computing system, such as a network endpoint system, may be to permit authorized system access since unauthorized access to the system can result in degraded performance, loss of service to authorized clients, loss of content on the system, etc. Network endpoint systems may include a wide variety of computing devices, including but not limited to, classic general purpose servers, network appliances (specialized servers), content delivery systems, home or laptop computers, clients, any other device that operates as an endpoint network connection, etc.

[0004] In computer network parlance, "security" is a catchall term that refers to the task of ensuring that only authorized clients are allowed access to a computing system and of preventing unauthorized intrusions or attacks. The security counter measures implemented with network connected computing systems are often denoted as firewalls. Network endpoint systems may be implemented with a stand alone hardware based firewall located between the endpoint system and the external network to which the endpoint system is coupled. A hardware based firewall often is expensive, inflexible and a performance bottleneck. Alternatively, a software based firewall often has more flexibility but is even slower than hardware based solutions.

[0005] There are many different types of security attacks and different types of attacks require different security counter measures. Many software programming techniques have been developed to detect security attacks and to prevent them from having adverse effects on an endpoint computer system. These software techniques are referred to as "security tools" and may be implemented in the hardware or software based firewall.

[0006] A common example of a network security tool is one that detects "denial of service" (DoS) attacks. A DoS attack is an attack on a system with the intent of preventing clients from using the attacked server. Generally, a DoS attack is not an attempt to gain unauthorized access to a system; rather, its objective is to prevent legitimate clients from gaining authorized access by overwhelming the system with connection requests.

[0007] DoS attacks make use of weaknesses in network protocols, such as the TCP/IP protocol. For example, if a sufficient number of new TCP connection requests are sent to an endpoint system (such as a server or content delivery system) at the same time, the endpoint system will attempt to establish a connection for each request. Eventually, the system's connection table fills up or other resources become depleted, which can result in loss of system functionality.

[0008] Many other types of attacks on network connected computing systems are also known. For example in "ping" attacks, a ping request is received with a broadcast destination request resulting in massive amounts of replies to the ping. A "Trojan horse" attack may execute a program on the endpoint computer system and cause an unauthorized transfer of data to the external network. In a "syn" attack, syn requests may cause the TCP/IP stack to overfill. Other attacks may rely on the use of "bogus" source addresses. Security tools often address these types of attacks (and others) and may further include other security functions. Such additional security functions commonly required of a network security system include authentication verification, packet filtering, and access control list (ACL) enforcement.

[0009] Detection of attempted security violations often requires the endpoint computing system to examine packets to distinguish real requests from attack based requests. The packet processing may also require comparisons among packets in a series, to detect various attack "signatures". All of this processing requires use of processor and memory resources. For this reason security measures are often implemented as a standalone network device.

SUMMARY OF THE INVENTION

[0010] One aspect of the invention is a network processing system connected to a network that carries data in packet format. Intelligent security hardware is placed at the network interface of the endpoint system. The intelligent security hardware (or a security accelerator) is programmed to receive packets from the network and to examine each packet to determine whether data in the packet represents a potential security violation. One or more processing units are programmed to execute some form of application and/or storage programming, and to thereby respond to requests contained within the packets. An interconnection medium is used to directly connect the security hardware to the processing units.

[0011] An advantage of the invention is that the security hardware at the front end of the endpoint eliminates the need for a firewall. Security tools are offloaded from other endpoint resources to the security hardware, so that the other resources can be devoted to server application tasks. The security hardware has the capability of performing "look ahead" Ad processing, thus unburdening the processing units that must perform the basic tasks appropriate for that particular network node.

[0012] The security hardware may be implemented with either a network processor or a CPU type general processor or a combination thereof. When a network processor is used, it performs "pass through" type processing that especially suitable for many types of security algorithms. This type of processing can be more suited for these algorithms than the state-modification intensive and memory-access intensive processing of a CPU type processor.

[0013] The security hardware can detect attempted security breaches very quickly. It can take immediate action,

such as discarding the packet or notifying the network administrator. Its programming can be easily updated by means of a simple download, making it easy to upgrade the security hardware to detect new types of attacks.

DESCRIPTION OF THE FIGURES

[0014] **FIG. 1A** is a representation of components of a content delivery system according to one embodiment of the disclosed content delivery system.

[0015] **FIG. 1B** is a representation of data flow between modules of a content delivery system of **FIG. 1A** according to one embodiment of the disclosed content delivery system.

[0016] **FIG. 1C** is a simplified schematic diagram showing one possible network content delivery system hardware configuration.

[0017] **FIG. 1D** is a simplified schematic diagram showing a network content delivery engine configuration possible with the network content delivery system hardware configuration of FIG.

[0018] **FIG. 1E** is a simplified schematic diagram showing an alternate network content delivery engine configuration possible with the network content delivery system hardware configuration of **FIG. 1C**.

[0019] **FIG. 1F** is a simplified schematic diagram showing another alternate network content delivery engine configuration possible with the network content delivery system hardware configuration of **FIG. 1C**.

[0020] FIGS. 1G-1J illustrate exemplary clusters of network content delivery systems.

[0021] **FIG. 2** is a simplified schematic diagram showing another possible network content delivery system configuration.

[0022] **FIG. 2A** is a simplified schematic diagram showing a network endpoint computing system.

[0023] **FIG. 2B** is a simplified schematic diagram showing a network endpoint computing system.

[0024] **FIG. 3** is a functional block diagram of an exemplary network processor.

[0025] **FIG. 4** is a functional block diagram of an exemplary interface between a switch fabric and a processor.

[0026] FIGS. **5-8** illustrate various systems having a network security accelerator in accordance with the invention.

DETAILED DESCRIPTION

[0027] Disclosed herein are systems and methods for operating network connected computing systems. The network connected computing systems disclosed provide a more efficient use of computing system resources and provide improved performance as compared to traditional network connected computing systems. Network connected computing systems may include network endpoint systems. The systems and methods disclosed herein may be particularly Otis beneficial for use in network endpoint systems. Network endpoint systems may include a wide variety of computing devices, including but not limited to, classic general purpose servers, specialized servers, network appliances, storage area networks or other storage medium, content delivery systems, corporate data centers, application service providers, home or laptop computers, clients, any other device that operates as an endpoint network connection, etc.

[0028] Other network connected systems may be considered a network intermediate node system. Such systems are generally connected to some node of a network that may operate in some other fashion than an endpoint. Typical examples include network switches or network routers. Network intermediate node systems may also include any other devices coupled to intermediate nodes of a network.

[0029] Further, some devices may be considered both a network intermediate node system and a network endpoint system. Such hybrid systems may perform both endpoint functionality and intermediate node functionality in the same device. For example, a network switch that also performs some endpoint functionality may be considered a hybrid system. As used herein such hybrid devices are considered to be a network endpoint system and are also considered to be a network intermediate node system.

[0030] For ease of understanding, the systems and methods disclosed herein are described with regards to an illustrative network connected computing system. In the illustrative example the system is a network endpoint system optimized for a content delivery application. Thus a content delivery system is provided as an illustrative example that demonstrates the structures, methods, advantages and benefits of the network computing system and methods disclosed herein. Content delivery systems (such as systems for serving streaming content, HTTP content, cached content, etc.) generally have intensive input/output demands.

[0031] It will be recognized that the hardware and methods discussed below may be incorporated into other hardware or applied to other applications. For example with respect to hardware, the disclosed system and methods may be utilized in network switches. Such switches may be considered to be intelligent or smart switches with expanded functionality beyond a traditional switch. Referring to the content delivery application described in more detail herein, a network switch may be configured to also deliver at least some content in addition to traditional switching functionality. Thus, though the system may be considered primarily a network switch (or some other network intermediate node device), the system may incorporate the hardware and methods disclosed herein. Likewise a network switch performing applications other than content delivery may utilize the systems and methods disclosed herein. The nomenclature used for devices utilizing the concepts of the present invention may vary. The network switch or router that includes the content delivery system disclosed herein may be called a network content switch or a network content router or the like. Independent of the nomenclature assigned to a device, it will be recognized that the network device may incorporate some or all of the concepts disclosed herein.

[0032] The disclosed hardware and methods also may be utilized in storage area networks, network attached storage, channel attached storage systems, disk arrays, tape storage systems, direct storage devices or other storage systems. In this case, a storage system having the traditional storage system functionality may also include additional functionality utilizing the hardware and methods shown herein.

Thus, although the system may primarily be considered a storage system, the system may still include the hardware and methods disclosed herein. The disclosed hardware and methods of the present invention also may be utilized in traditional personal computers, portable computers, servers, workstations, mainframe computer systems, or other computer systems. In this case, a computer system having the traditional computer system functionality associated with the particular type of computer system may also include additional functionality utilizing the hardware and methods shown herein. Thus, although the system may primarily be considered to be a particular type of computer system, the system may still include the hardware and methods disclosed herein.

[0033] As mentioned above, the benefits of the present invention are not limited to any specific tasks or applications. The content delivery applications described herein are thus illustrative only. Other tasks and applications that may incorporate the principles of the present invention include, but are not limited to, database management systems, application service providers, corporate data centers, modeling and simulation systems, graphics rendering systems, other complex computational analysis systems, etc. Although the principles of the present invention may be described with respect to a specific application, it will be recognized that many other tasks or applications performed with the hardware and methods.

[0034] Disclosed herein are systems and methods for delivery of content to computer-based networks that employ functional multi-processing using a "staged pipeline" content delivery environment to optimize bandwidth utilization and accelerate content delivery while allowing greater determination in the data traffic management. The disclosed systems may employ individual modular processing engines that are optimized for different layers of a software stack. Each individual processing engine may be provided with one or more discrete subsystem modules configured to run on their own optimized platform and/or to function in parallel with one or more other subsystem modules across a high speed distributive interconnect, such as a switch fabric, that allows peer-to-peer communication between individual subsystem modules. The use of discrete subsystem modules that are distributively interconnected in this manner advantageously allows individual resources (e.g., processing resources, memory resources) to be deployed by sharing or reassignment in order to maximize acceleration of content delivery by the content delivery system. The use of a scalable packet-based interconnect, such as a switch fabric, advantageously allows the installation of additional subsystem modules without significant degradation of system performance. Furthermore, policy enhancement/enforcement may be optimized by placing intelligence in each individual modular processing engine.

[0035] The network systems disclosed herein may operate as network endpoint systems. Examples of network endpoints include, but are not limited to, servers, content delivery systems, storage systems, application service providers, database management systems, corporate data center servers, etc. A client system is also a network endpoint, and its resources may typically range from those of a general purpose computer to the simpler resources of a network appliance. The various processing units of the network endpoint system may be programmed to achieve the desired type of endpoint.

[0036] Some embodiments of the network endpoint systems disclosed herein are network endpoint content delivery systems. The network endpoint content delivery systems may be m utilized in replacement of or in conjunction with traditional network servers. A "server" can be any device that delivers content, services, or both. For example, a content delivery server receives requests for content from remote browser clients via the network, accesses a file system to retrieve the requested content, and delivers the content to the client. As another example, an applications server may be programmed to execute applications software on behalf of a remote client, thereby creating data for use by the client. Various server appliances are being developed and often perform specialized tasks.

[0037] As will be described more fully below, the network endpoint system disclosed herein may include the use of network processors. Though network processors conventionally are designed and utilized at intermediate network nodes, the network endpoint system disclosed herein adapts this type of processor for endpoint use.

[0038] The network endpoint system disclosed may be construed as a switch based computing system. The system may further be characterized as an asymmetric multi-processor system configured in a staged pipeline manner.

[0039] Exemplary System Overview

[0040] **FIG. 1A** is a representation of one embodiment of a content delivery system **1010**, for example as may be employed as a network endpoint system in connection with a network **1020**. Network **1020** may be any type of computer network suitable for linking computing systems. Content delivery system **1010** may be coupled to one or more networks including, but not limited to, the public internet, a private intranet network (e.g., linking users and hosts such as employees of a corporation or institution), a wide area network (WAN), a local area network (LAN), a wireless network, any other client based network or any other network environment of connected computer systems or online users. Thus, the data provided from the network **1020** may be in any networking protocol. In one embodiment, network **1020** may be the public internet that serves to provide access to content delivery system **1010** by multiple online users that utilize internet web browsers on personal computers operating through an internet service provider. In this case the data is assumed to follow one or more of various Internet Protocols, such as TCP/IP, UDP, HTTP, RTSP, SSL, FTP, etc. However, the same concepts apply to networks using other existing or future protocols, such as IPX, SNMP, NetBios, Ipv6, etc. The concepts may also apply to file protocols such as network file system (NFS) or common internet file system (CIFS) file sharing protocol.

[0041] Examples of content that may be delivered by content delivery system **1010** include, but are not limited to, static content (e.g., web pages, MP3 files, HTTP object files, audio stream files, video stream files, etc.), dynamic content, etc. In this regard, static content may be defined as content available to content delivery system **1010** via attached storage devices and as content that does not generally require any processing before delivery. Dynamic content, on

the other hand, may be defined as content that either requires processing before delivery, or resides remotely from content delivery system **1010**. As illustrated in **FIG. 1A**, content sources may include, but are not limited to, one or more storage devices **1090** (magnetic disks, optical disks, tapes, storage area networks (SAN's), etc.), other content sources **1100**, third party remote content feeds, broadcast sources (live direct audio or video broadcast feeds, etc.), delivery of cached content, combinations thereof, etc. Broadcast or remote content may be advantageously received through second network connection **1023** and delivered to network **1020** via an accelerated flowpath through content delivery system **1010**. As discussed below, second network connection **1023** may be connected to a second network **1024** (as shown). Alternatively, both network connections **1022** and **1023** may be connected to network **1020**.

[**0042**] As shown in **FIG. 1A**, one embodiment of content delivery system **1010** includes multiple system engines **1030, 1040, 1050, 1060,** and **1070** communicatively coupled via distributive interconnection **1080**. In the exemplary embodiment provided, these system engines operate as content delivery engines. As used herein, "content delivery engine" generally includes any hardware, software or hardware/software combination capable of performing one or more #He dedicated tasks or sub-tasks associated with the delivery or transmittal of content from one or more content sources to one or more networks. In the embodiment illustrated in **FIG. 1A** content delivery processing engines (or "processing blades") include network interface processing engine **1030**, storage processing engine **1040**, network transport/protocol processing engine **1050** (referred to hereafter as a transport processing engine), system management processing engine **1060**, and application processing engine **1070**. Thus configured, content delivery system **1010** is capable of providing multiple dedicated and independent processing engines that are optimized for networking, storage and application protocols, each of which is substantially self-contained and therefore capable of functioning without consuming resources of the remaining processing engines.

[**0043**] It will be understood with benefit of this disclosure that the particular number and identity of content delivery engines illustrated in **FIG. 1A** are illustrative only, and that for any given content delivery system **1010** the number and/or identity of content delivery engines may be varied to fit particular needs of a given application or installation. Thus, the number of engines employed in a given content delivery system may be greater or fewer in number than illustrated in **FIG. 1A**, and/or the selected engines may include other types of content delivery engines and/or may not include all of the engine types illustrated in **FIG. 1A**. In one embodiment, the content delivery system **1010** may be implemented within a single chassis, such as for example, a 2U chassis.

[**0044**] Content delivery engines **1030, 1040, 1050, 1060** and **1070** are present to independently perform selected sub-tasks associated with content delivery from content sources **1090** and/or **1100**, it being understood however that in other embodiments any one or more of such subtasks may be combined and performed by a single engine, or subdivided to be performed by more than one engine. In one embodiment, each of engines **1030, 1040, 1050, 1060** and **1070** may employ one or more independent processor modules (e.g., CPU modules) having independent processor and memory subsystems and suitable for performance of a given functions, allowing independent operation without interference from other engines or modules. Advantageously, this allows custom selection of particular processor-types based on the particular sub-task each is to perform, and in consideration of factors such as speed or efficiency in performance of a given subtask, cost of individual processor, etc. The processors utilized may be any processor suitable for adapting to endpoint processing. Any "PC on a board" type device may be used, such as the x86 and Pentium processors from Intel Corporation, the SPARC processor from Sun Microsystems, Inc., the PowerPC processor from Motorola, Inc. or any other microcontroller or microprocessor. In addition, network processors (discussed in more detail below) may also be utilized. The modular multi-task configuration of content delivery system **1010** allows the number and/or type of content delivery engines and processors to be selected or varied to fit the needs of a particular application.

[**0045**] The configuration of the content delivery system described above provides scalability without having to scale all the resources of a system. Thus, unlike the traditional rack and stack systems, such as server systems in which an entire server may be added just to expand one segment of system resources, the content delivery system allows the particular resources needed to be the only expanded resources. For example, storage resources may be greatly expanded without having to expand all of the traditional server resources.

[**0046**] Distributive Interconnect

[**0047**] Still referring to **FIG. 1A**, distributive interconnection **1080** may be any multi-node I/O interconnection hardware or hardware/software system suitable for distributing functionality by selectively interconnecting two or more content delivery engines of a content delivery system including, but not limited to, high speed interchange systems such as a switch fabric or bus architecture. Examples of switch fabric architectures include cross-bar switch fabrics, Ethernet switch fabrics, ATM switch fabrics, etc. Examples of bus architectures include PCI, PCI-X, S-Bus, Microchannel, VME, etc. Generally, for purposes of this description, a "bus" is any system bus that carries data in a manner that is visible to all nodes on the bus. Generally, some sort of bus arbitration scheme is implemented and data may be carried in parallel, as n-bit words. As distinguished from a bus, a switch fabric establishes independent paths from node to node and data is specifically addressed to a particular node on the switch fabric. Other nodes do not see the data nor are they blocked from creating their own paths. The result is a simultaneous guaranteed bit rate in each direction for each of the switch fabric's ports.

[**0048**] The use of a distributed interconnect **1080** to connect the various processing engines in lieu of the network connections used with the switches of conventional multi-server endpoints is beneficial for several reasons. As compared to network connections, the distributed interconnect **1080** is less error prone, allows more deterministic content delivery, and provides higher bandwidth connections to the various processing engines. The distributed interconnect **1080** also has greatly improved data integrity and throughput rates as compared to network connections.

[**0049**] Use of the distributed interconnect **1080** allows latency between content delivery engines to be short, finite

and follow a known path. Known maximum latency specifications are typically associated with the various bus architectures listed above. Thus, when the employed interconnect medium is a bus, latencies fall within a known range. In the case of a switch fabric, latencies are fixed. Further, the connections are "direct", rather than by some undetermined path. In general, the use of the distributed interconnect **1080** rather than network connections, permits the switching and interconnect capacities of the content delivery system **1010** to be predictable and consistent.

[0050] One example interconnection system suitable for use as distributive interconnection **1080** is an 8/16 port 28.4 Gbps high speed PRIZMA-E non-blocking switch fabric switch available from IBM. It will be understood that other switch fabric configurations having greater or lesser numbers of ports, throughput, and capacity are also possible. Among the advantages offered by such a switch fabric interconnection in comparison to shared-bus interface interconnection technology are throughput, scalability and fast and efficient communication between individual discrete content delivery engines of content delivery system **1010**. In the embodiment of **FIG. 1A**, distributive interconnection **1080** facilitates parallel and independent operation of each engine in its own optimized environment without bandwidth interference from other engines, while at the same time providing peer-to-peer communication between the engines on an as-needed basis (e.g., allowing direct communication between any two content delivery engines **1030**, **1040**, **1050**, **1060** and **1070**). Moreover, the distributed interconnect may directly transfer inter-processor communications between the various engines of the system. Thus, communication, command and control information may be provided between the various peers via the distributed interconnect. In addition, communication from one peer to multiple peers may be implemented through a broadcast communication which is provided from one peer to all peers coupled to the interconnect. The interface for each peer may be standardized, thus providing ease of design and allowing for system scaling by providing standardized ports for adding additional peers.

[0051] Network Interface Processing Engine

[0052] As illustrated in **FIG. 1A**, network interface processing engine **1030** interfaces with network **1020** by receiving and processing requests for content and delivering requested content to network **1020**. Network interface processing engine **1030** may be any hardware or hardware/software subsystem suitable for connections utilizing TCP (Transmission Control Protocol) IP (Internet Protocol), UDP (User Datagram Protocol), RTP (Real-Time Transport Protocol), Internet Protocol (IP), Wireless Application Protocol (WAP) as well as other networking protocols. Thus the network interface processing engine **1030** may be suitable for handling queue management, buffer management, TCP connect sequence, checksum, IP address lookup, internal load balancing, packet switching, etc. Thus, network interface processing engine **1030** may be employed as illustrated to process or terminate one or more layers of the network protocol stack and to perform look-up intensive operations, offloading these tasks from other content delivery processing engines of content delivery system **1010**. Network interface processing engine **1030** may also be employed to load balance among other content delivery processing engines of content delivery system **1010**. Both of these features serve to accelerate content delivery, and are enhanced by placement of distributive interchange and protocol termination processing functions on the same board. Examples of other functions that may be performed by network interface processing engine **1030** include, but are not limited to, security processing.

[0053] With regard to the network protocol stack, the stack in traditional systems may often be rather large. Processing the entire stack for every request across the distributed interconnect may significantly impact performance. As described herein, the protocol stack has been segmented or "split" between the network interface engine and the transport processing engine. An abbreviated version of the protocol stack is then provided across the interconnect. By utilizing this functionally split version of the protocol stack, increased bandwidth may be obtained. In this manner the communication and data flow through the content delivery system **1010** may be accelerated. The use of a distributed interconnect (for example a switch fabric) further enhances this acceleration as compared to traditional bus interconnects.

[0054] The network interface processing engine **1030** may be coupled to the network **1020** through a Gigabit (Gb) Ethernet fiber front end interface **1022**. One or more additional Gb Ethernet interfaces **1023** may optionally be provided, for example, to form a second interface with network **1020**, or to form an interface with a second network or application **1024** as shown (e.g., to form an interface with one or more server/s for delivery of web cache content, etc.). Regardless of whether the network connection is via Ethernet, or some other means, the network connection could be of any type, with other examples being ATM, SONET, or wireless. The physical medium between the network and the network processor may be copper, optical fiber, wireless, etc.

[0055] In one embodiment, network interface processing engine **1030** may utilize a network processor, although it will be understood that in other embodiments a network processor may be supplemented with or replaced by a general purpose processor or an embedded microcontroller. The network processor may be one of the various types of specialized processors that have been designed and marketed to switch network traffic at intermediate nodes. Consistent with this conventional application, these processors are designed to process high speed streams of network packets. In conventional operation, a network processor receives a packet from a port, verifies fields in the packet header, and decides on an outgoing port to which it forwards the packet. The processing of a network processor may be considered as "pass through" processing, as compared to the intensive state modification processing performed by general purpose processors. A typical network processor has a number of processing elements, some operating in parallel and some in pipeline. Often a characteristic of a network processor is that it may hide memory access latency needed to perform lookups and modifications of packet header fields. A network processor may also have one or more network interface controllers, such as a gigabit Ethernet controller, and are generally capable of handling data rates at "wire speeds".

[0056] Examples of network processors include the C-Port processor manufactured by Motorola, Inc., the IXP1200 processor manufactured by Intel Corporation, the Prism

processor manufactured by SiTera Inc., and others manufactured by MMC Networks, Inc. and Agere, Inc. These processors are programmable, usually with a RISC or augmented RISC instruction set, and are typically fabricated on a single chip.

[0057] The processing cores of a network processor are typically accompanied by special purpose cores that perform specific tasks, such as fabric interfacing, table lookup, queue management, and buffer management. Network processors typically have their memory management optimized for data movement, and have multiple I/O and memory buses. The programming capability of network processors permit them to be programmed for a variety of tasks, such as load balancing, network protocol processing, network security policies, and QoS/CoS support. These tasks can be tasks that would otherwise be performed by another processor. For example, TCP/IP processing may be performed by a network processor at the front end of an endpoint system. Another type of processing that could be offloaded is execution of network security policies or protocols. A network processor could also be used for load balancing. Network processors used in this manner can be referred to as "network accelerators" because their front end "look ahead" processing can vastly increase network response speeds. Network processors perform look ahead processing by operating at the front end of the network endpoint to process network packets in order to reduce the workload placed upon the remaining endpoint resources. Various uses of network accelerators are described in the following concurrently filed U.S. patent applications No. _____, entitled "Network Transport Accelerator," by Bailey et. al; No. _____ entitled "Single Chassis Network Endpoint System With Network Processor For Load Balancing," by Richter et. al; the disclosures of which are all incorporated herein by reference. When utilizing network processors in an endpoint environment it may be advantageous to utilize techniques for order serialization of information, such as for example, as disclosed in concurrently filed U.S. patent application No. _____, entitled "Methods and Systems For The Order Serialization Of Information In A Network Processing Environment," by Richter et. al, the disclosure of which is incorporated herein by reference.

[0058] **FIG. 3** illustrates one possible general configuration of a network processor. As illustrated, a set of traffic processors **21** operate in parallel to handle transmission and receipt of network traffic. These processors may be general purpose microprocessors or state machines. Various core processors **22-24** handle special tasks. For example, the core processors **22-24** may handle lookups, checksums, and buffer management. A set of serial data processors **25** provide Layer 1 network support. Interface **26** provides the physical interface to the network **1020**. A general purpose bus interface **27** is used for downloading code and configuration tasks. A specialized interface **28** may be specially programmed to optimize the path between network processor **12** and distributed interconnection **1080**.

[0059] As mentioned above, the network processors utilized in the content delivery system **1010** are utilized for endpoint use, rather than conventional use at intermediate network nodes. In one embodiment, network interface processing engine **1030** may utilize a MOTOROLA C-Port C-5 network processor capable of handling two Gb Ethernet interfaces at wire speed, and optimized for cell and packet processing. This network processor may contain sixteen 200 MHz MIPS processors for cell/packet switching and thirty-two serial processing engines for bit/byte processing, checksum generation/verification, etc. Further processing capability may be provided by five co-processors that perform the following network specific tasks: supervisor/executive, switch fabric interface, optimized table lookup, queue management, and buffer management. The network processor may be coupled to the network **1020** by using a VITESSE GbE SERDES (serializer-deserializer) device (for example the VSC7123) and an SFP (small form factor pluggable) optical transceiver for LC fiber connection.

[0060] Transport/Protocol Processing Engine

[0061] Referring again to **FIG. 1A**, transport processing engine **1050** may be provided for performing network transport protocol sub-tasks, such as processing content requests received from network interface engine **1030**. Although named a "transport" engine for discussion purposes, it will be recognized that the engine **1050** performs transport and protocol processing and the term transport processing engine is not meant to limit the functionality of the engine. In this regard transport processing engine **1050** may be any hardware or hardware/software subsystem suitable for TCP/UDP processing, other protocol processing, transport processing, etc. In one embodiment transport engine **1050** may be a dedicated TCP/UDP processing module alp based on an INTEL PENTIUM III or MOTOROLA POWERPC 7450 based processor running the Thread-X RTOS environment with protocol stack based on TCP/IP technology.

[0062] As compared to traditional server type computing systems, the transport processing engine **1050** may off-load other tasks that traditionally a main CPU may perform. For example, the performance of server CPUs significantly decreases when a large amount of network connections are made merely because the server CPU regularly checks each connection for time outs. The transport processing engine **1050** may perform time out checks for each network connection, session management, data reordering and retransmission, data queueing and flow control, packet header generation, etc. off-loading these tasks from the application processing engine or the network interface processing engine. The transport processing engine **1050** may also handle error checking, likewise freeing up the resources of other processing engines.

[0063] Network Interface/Transport Split Protocol

[0064] The embodiment of **FIG. 1A** contemplates that the protocol processing is shared between the transport processing engine **1050** and the network interface engine **1030**. This sharing technique may be called "split protocol stack" processing. The division of tasks may be such that higher tasks in the protocol stack are assigned to the transport processor engine. For example, network interface engine **1030** may processes all or some of the TCP/IP protocol stack as well as all protocols lower on the network protocol stack. Another approach could be to assign state modification intensive tasks to the transport processing engine.

[0065] In one embodiment related to a content delivery system that receives packets, the network interface engine performs the MAC header identification and verification, IP header identification and verification, IP header checksum

validation, TCP and UDP header identification and validation, and TCP or UDP checksum validation. It also may perform the lookup to determine the TCP connection or UDP socket (protocol session identifier) to which a received packet belongs. Thus, the network interface engine verifies packet lengths, checksums, and validity. For transmission of packets, the network interface engine performs TCP or UDP checksum generation, IP header generation, and MAC header generation, IP checksum generation, MAC FCS/CRC generation, etc.

[0066] Tasks such as those described above can all be performed rapidly by the parallel and pipeline processors within a network processor. The "fly by" processing style of a network processor permits it to look at each byte of a packet as it passes through, using registers and other alternatives to memory access. The network processor's "stateless forwarding" operation is best suited for tasks not involving complex calculations that require rapid updating of state information.

[0067] An appropriate internal protocol may be provided for exchanging information between the network interface engine **1030** and the transport engine **1050** when setting up or terminating a TCP and/or UDP connections and to transfer packets between the two engines. For example, where the distributive interconnection medium is a switch fabric, the internal protocol may be implemented as a set of messages exchanged across the switch fabric. These messages indicate the arrival of new inbound or outbound connections and contain inbound or outbound packets on existing connections, along with identifiers or tags for those connections. The internal protocol may also be used to transfer identifiers or tags between the transport engine **1050** and the application processing engine **1070** and/or the storage processing engine **1040**. These identifiers or tags may be used to reduce or strip or accelerate a portion of the protocol stack.

[0068] For example, with a TCP/IP connection, the network interface engine **1030** may receive a request for a new connection. The header information associated with the initial request may be provided to the transport processing engine **1050** for processing. That result of this processing may be stored in the resources of the transport processing engine **1050** as state and management information for that particular network session. The transport processing engine **1050** then informs the network interface engine **1030** as to the location of these results. Subsequent packets related to that connection that are processed by the network interface engine **1030** may have some of the header information stripped and replaced with an identifier or tag that is provided to the transport processing engine **1050**. The identifier or tag may be a pointer, index or any other mechanism that provides for the identification of the location in the transport processing engine of the previously setup state and management information (or the corresponding network session). In this manner, the transport processing engine **1050** does not have to process the header information of every packet of a connection. Rather, the transport interface engine merely receives a contextually meaningful identifier or tag that identifies the previous processing results for that connection.

[0069] In one embodiment, the data link, network, transport and session layers (layers 2-5) of a packet may be replaced by identifier or tag information. For packets related to an established connection the transport processing engine does not have to perform intensive processing with regard to these layers such as hashing, scanning, look up, etc. operations. Rather, these layers have already been converted (or processed) once in the transport processing engine and the transport processing engine just receives the identifier or tag provided from the network interface engine that identifies the location of the conversion results.

[0070] In this manner an identifier or tag is provided for each packet of an established connection so that the more complex data computations of converting header information may be replaced with a more simplistic analysis of an identifier or tag. The delivery of content is thereby accelerated, as the time for packet processing and the amount of system resources for packet processing are both reduced. The functionality of network processors, which provide efficient parallel processing of packet headers, is well suited for enabling the acceleration described herein. In addition, acceleration is further provided as the physical size of the packets provided across the distributed interconnect may be reduced.

[0071] Though described herein with reference to messaging between the network interface engine and the transport processing engine, the use of identifiers or tags may be utilized amongst all the engines in the modular pipelined processing described herein. Thus, one engine may replace packet or data information with contextually meaningful information that may require less processing by the next engine in the data and communication flow path. In addition, these techniques may be utilized for a wide variety of protocols and layers, not just the exemplary embodiments provided herein.

[0072] With the above-described tasks being performed by the network interface engine, the transport engine may perform TCP sequence number processing, acknowledgement and go retransmission, segmentation and reassembly, and flow control tasks. These tasks generally call for storing and modifying connection state information on each TCP and UDP connection, and therefore are considered more appropriate for the processing capabilities of general purpose processors.

[0073] As will be discussed with references to alternative embodiments (such as **FIGS. 2 and 2A**), the transport engine **1050** and the network interface engine **1030** may be combined into a single engine. Such a combination may be advantageous as communication across the switch fabric is not necessary for protocol processing. However, limitations of many commercially available network processors make the split protocol stack processing described above desirable.

[0074] Application Processing Engine

[0075] Application processing engine **1070** may be provided in content delivery system **1010** for application processing, and may be, for example, any hardware or hardware/software subsystem suitable for session layer protocol processing (e.g., HTTP, RTSP streaming, etc.) of content requests received from network transport processing engine **1050**. In one embodiment application processing engine **1070** may be a dedicated application processing module based on an INTEL PENTIUM III processor running, for example, on standard x86 OS systems (e.g., Linux, Win-

dows NT, FreeBSD, etc.). Application processing engine **1070** may be utilized for dedicated application-only processing by virtue of the off-loading of all network protocol and storage processing elsewhere in content delivery system **1010**. In one embodiment, processor programming for application processing engine **1070** may be generally similar to that of a conventional server, but without the tasks off-loaded to network interface processing engine **1030**, storage processing engine **1040**, and transport processing engine **1050**.

[0076] Storage Management Engine

[0077] Storage management engine **1040** may be any hardware or hardware/software subsystem suitable for effecting delivery of requested content from content sources (for example content sources **1090** and/or **1100**) in response to processed requests received from application processing engine **1070**. It will also be understood that in various embodiments a storage management engine **1040** may be employed with content sources other than disk drives (e.g., solid state storage, the storage systems described above, or any other media suitable for storage of data) and may be programmed to request and receive data from these other types of storage.

[0078] In one embodiment, processor programming for storage management engine **1040** may be optimized for data retrieval using techniques such as caching, and may include and maintain a disk cache to reduce the relatively long time often required to retrieve data from content sources, such as disk drives. Requests received by storage management engine **1040** from application processing engine **1070** may contain information on how requested data is to be formatted and its destination, with this information being comprehensible to transport processing engine **1050** and/or network interface processing engine **1030**. The storage management engine **1040** may utilize a disk cache to reduce the relatively long time it may take to retrieve data stored in a storage medium such as disk drives. Upon receiving a request, storage management engine **1040** may be programmed to first determine whether the requested data is cached, and then to send a request for data to the appropriate content source **1090** or **1100**. Such a request may be in the form of a conventional read request. The designated content source **1090** or **1100** responds by sending the requested content to storage management engine **1040**, which in turn sends the content to transport processing engine **1050** for forwarding to network interface processing engine **1030**.

[0079] Based on the data contained in the request received from application processing engine **1070**, storage processing engine **1040** sends the requested content in proper format with the proper destination data included. Direct communication between storage processing engine **1040** and transport processing engine **1050** enables application processing engine **1070** to be bypassed with the requested content. Storage processing engine **1040** may also be configured to write data to content sources **1090** and/or **1100** (e.g., for storage of live or broadcast streaming content).

[0080] In one embodiment storage management engine **1040** may be a dedicated block-level cache processor capable of block level cache processing in support of thousands of concurrent multiple readers, and direct block data switching to network interface engine **1030**. In this regard storage management engine **1040** may utilize a POWER PC 7450 processor in conjunction with ECC memory and a LSI SYMFC929 dual 2 GBaud fibre channel controller for fibre channel interconnect to content sources **1090** and/or **1100** via dual fibre channel arbitrated loop **1092**. It will be recognized, however, that other forms of interconnection to storage sources suitable for retrieving content are also possible. Storage management engine **1040** may include hardware and/or software for running the Fibre Channel (FC) protocol, the SCSI (Small Computer Systems Interface) protocol, iSCSI protocol as well as other storage networking protocols.

[0081] Storage management engine **1040** may employ any suitable method for caching data, including simple computational caching algorithms such as random removal (RR), first-in first-out (FIFO), predictive read-ahead, over buffering, etc. algorithms. Other suitable caching algorithms include those that consider one or more factors in the manipulation of content stored within the cache memory, or which employ multi-level ordering, key based ordering or function based calculation for replacement. In one embodiment, storage management engine may implement a layered multiple LRU (LMLRU) algorithm that uses an integrated block/buffer management structure including at least two layers of a configurable number of multiple LRU queues and a two-dimensional positioning algorithm for data blocks in the memory to reflect the relative priorities of a data block in the memory in terms of both recency and frequency. Such a caching algorithm is described in further detail in concurrently filed U.S. patent application No. _____, entitled "Systems and Methods for Management of Memory" by Qiu et. al, the disclosure of which is incorporated herein by reference.

[0082] For increasing delivery efficiency of continuous content, such as streaming multimedia content, storage management engine **1040** may employ caching algorithms that consider the dynamic characteristics of continuous content. Suitable examples include, but are not limited to, interval caching algorithms. In one embodiment, improved caching performance of continuous content may be achieved using an LMLRU caching algorithm that weighs ongoing viewer cache value versus the dynamic time-size cost of maintaining particular content in cache memory. Such a caching algorithm is described in further detail in concurrently filed U.S. patent application No. _____, entitled "Systems and Methods for Management of Memory in Information Delivery Environments" by Qiu et. al, the disclosure of which is incorporated herein by reference.

[0083] System Management Engine

[0084] System management (or host) engine **1060** may be present to perform system management functions related to the operation of content delivery system **1010**. Examples of system management functions include, but are not limited to, content provisioning/updates, comprehensive statistical data gathering and logging for sub-system engines, collection of shared user bandwidth utilization and content utilization data that may be input into billing and accounting systems, "on the fly" ad insertion into delivered content, customer programmable sub-system level quality of service ("QoS") parameters, remote management (e.g., SNMP, web-based, CLI), health monitoring, clustering controls, remote/local disaster recovery functions, predictive performance and capacity planning, etc. In one embodiment, content

delivery bandwidth utilization by individual content suppliers or users (e.g., individual supplier/user usage of distributive interchange and/or content delivery engines) may be tracked and logged by system management engine **1060**, enabling an operator of the content delivery system **1010** to charge each content supplier or user on the basis of content volume delivered.

[0085] System management engine **1060** may be any hardware or hardware/software subsystem suitable for performance of one or more such system management engines and in one embodiment may be a dedicated application processing module based, for example, on an INTEL PENTIUM III processor running an x86 OS. Because system management engine **1060** is provided as a discrete modular engine, it may be employed to perform system management functions from within content delivery system **1010** without adversely affecting the performance of the system. Furthermore, the system management engine **1060** may maintain information on processing engine assignment and content delivery paths for various content delivery applications, substantially eliminating the need for an individual processing engine to have intimate knowledge of the hardware it intends to employ.

[0086] Under manual or scheduled direction by a user, system management processing engine **1060** may retrieve content from the network **1020** or from one or more external servers on a second network **1024** (e.g., LAN) using, for example, network file system (NFS) or common internet file system (CIFS) file sharing protocol. Once content is retrieved, the content delivery system may advantageously maintain an independent copy of the original content, and therefore is free to employ any file system structure that is beneficial, and need not understand low level disk formats of a large number of file systems.

[0087] Management interface **1062** may be provided for interconnecting system management engine **1060** with a network **1200** (e.g., LAN), or connecting content delivery system **1010** to other network appliances such as other content delivery systems **1010**, servers, computers, etc. Management interface **1062** may be by any suitable network interface, such as 10/100 Ethernet, and may support communications such as management and origin traffic. Provision for one or more terminal management interfaces (not shown) for may also be provided, such as by RS-232 port, etc. The management interface may be utilized as a secure port to provide system management and control information to the content delivery system **1010**. For example, tasks which may be accomplished through the management interface **1062** include reconfiguration of the allocation of system hardware (as discussed below with reference to FIGS. 1C-1F), programming the application processing engine, diagnostic testing, and any other management or control tasks. Though generally content is not envisioned being provided through the management interface, the identification of or location of files or systems containing content may be received through the management interface **1062** so that the content delivery system may access the content through the other higher bandwidth interfaces.

[0088] Management Performed By The Network Interface

[0089] Some of the system management functionality may also be performed directly within the network interface processing engine **1030**. In this case some system policies and filters may be executed by the network interface engine **1030** in real-time at wirespeed. These polices and filters may manage some traffic/bandwidth management criteria and various service level guarantee policies. Examples of such system management functionality of are described below. It will be recognized that these functions may be performed by the system management engine **1060**, the network interface engine **1030**, or a combination thereof.

[0090] For example, a content delivery system may contain data for two web sites. An operator of the content delivery system may guarantee one web site ("the higher quality site") higher performance or bandwidth than the other web site ("the lower quality site"), presumably in exchange for increased compensation from the higher quality site. The network interface processing engine **1030** may be utilized to determine if the bandwidth limits for the lower quality site have been exceeded and reject additional data requests related to the lower quality site. Alternatively, requests related to the lower quality site may be rejected to ensure the guaranteed performance of the higher quality site is achieved. In this manner the requests may be rejected immediately at the interface to the external network and additional resources of the content delivery system need not be utilized. In another example, storage service providers may use the content delivery system to charge content providers based on system bandwidth of downloads (as opposed to the traditional storage area based fees). For billing purposes, the network interface engine may monitor the bandwidth use related to a content provider. The network interface engine may also reject additional requests related to content from a content provider whose bandwidth limits have been exceeded. Again, in this manner the requests may be rejected immediately at the interface to the external network and additional resources of the content delivery system need not be utilized.

[0091] Additional system management functionality, such as quality of service (QoS) functionality, also may be performed by the network interface engine. A request from the external network to the content delivery system may seek a specific file and also may contain Quality of Service (QoS) parameters. In one example, the QoS parameter may indicate the priority of service that a client on the external network is to receive. The network interface engine may recognize the QoS data and the data may then be utilized when managing the data and communication flow through the content delivery system. The request may be transferred to the storage management engine to access this file via a read queue, e.g., [Destination IP][Filename][File Type (CoS)][Transport Priorities (QoS)]. All file read requests may be stored in a read queue. Based on CoS/QoS policy parameters as well as buffer status within the storage management engine (empty, full, near empty, block seq#, etc), the storage management engine may prioritize which blocks of which files to access from the disk next, and transfer this data into the buffer memory location that has been assigned to be transmitted to a specific IP address. Thus based upon QoS data in the request provided to the content delivery system, the data and communication traffic through the system may be prioritized. The QoS and other policy priorities may be applied to both incoming and outgoing traffic flow. Therefore a request having a higher QoS priority may be received after a lower order priority request, yet the higher priority request may be served data before the lower priority request.

[0092] The network interface engine may also be used to filter requests that are not supported by the content delivery system. For example, if a content delivery system is configured only to accept HTTP requests, then other requests such as FTP, telnet, etc. may be rejected or filtered. This filtering may be applied directly at the network interface engine, for example by programming a network processor with the appropriate system policies. Limiting undesirable traffic directly at the network interface offloads such functions from the other processing modules and improves system performance by limiting the consumption of system resources by the undesirable traffic. It will be recognized that the filtering example described herein is merely exemplary and many other filter criteria or policies may be provided.

[0093] Multi-Processor Module Design

[0094] As illustrated in **FIG. 1A**, any given processing engine of content delivery system **1010** may be optionally provided with multiple processing modules so as to enable parallel or redundant processing of data and/or communications. For example, two or more individual dedicated TCP/UDP processing modules **1050***a* and **1050***b* may be provided for transport processing engine **1050**, two or more individual application processing modules **1070***a* and **1070***b* may be provided for network application processing engine **1070**, two or more individual network interface processing modules **1030***a* and **1030***b* may be provided for network interface processing engine **1030** and two or more individual storage management processing modules **1040***a* and **1040***b* may be provided for storage management processing engine **1040**. Using such a configuration, a first content request may be processed between a first TCP/UDP processing module and a first application processing module via a first switch fabric path, at the same time a second content request is processed between a second TCP/UDP processing module and a second application processing module via a second switch fabric path. Such parallel processing capability may be employed to accelerate content delivery.

[0095] Alternatively, or in combination with parallel processing capability, a first TCP/UDP processing module **1050***a* may be backed-up by a second TCP/UDP processing module **1050***b* that acts as an automatic failover spare to the first module **1050***a*. In those embodiments employing multiple-port switch fabrics, various combinations of multiple modules may be selected for use as desired on an individual system-need basis (e.g., as may be dictated by module failures and/or by anticipated or actual bottlenecks), limited only by the number of available ports in the fabric. This feature offers great flexibility in the operation of individual engines and discrete processing modules of a content delivery system, which may be translated into increased content delivery acceleration and reduction or substantial elimination of adverse effects resulting from system component failures.

[0096] In yet other embodiments, the processing modules may be specialized to specific applications, for example, for processing and delivering HTTP content, processing and delivering RTSP content, or other applications. For example, in such an embodiment an application processing module **1070***a* and storage processing module **1050***a* may be specially programmed for processing a first type of request received from a network. In the same system, application processing module **1070***b* and storage processing module **1050***b* may be specially programmed to handle a second type of request different from the first type. Routing of requests to the appropriate respective application and/or storage modules may be accomplished using a distributive interconnect and may be controlled by transport and/or interface processing modules as requests are received and processed by these modules using policies set by the system management engine.

[0097] Further, by employing processing modules capable of performing the function of more than one engine in a content delivery system, the assigned functionality of a given module may be changed on an as-needed basis, either manually or automatically by the system management engine upon the occurrence of given parameters or conditions. This feature may be achieved, for example, by using similar hardware modules for different content delivery engines (e.g., by employing PENTIUM III based processors for both network transport processing modules and for application processing modules), or by using different hardware modules capable of performing the same task as another module through software programmability (e.g., by employing a POWER PC processor based module for storage management modules that are also capable of functioning as network transport modules). In this regard, a content delivery system may be configured so that such functionality reassignments may occur during system operation, at system boot-up or in both cases. Such reassignments may be effected, for example, using software so that in a given content delivery system every content delivery engine (or at a lower level, every discrete content delivery processing module) is potentially dynamically reconfigurable using software commands. Benefits of engine or module reassignment include maximizing use of hardware resources to deliver content while minimizing the need to add expensive hardware to a content delivery system.

[0098] Thus, the system disclosed herein allows various levels of load balancing to satisfy a work request. At a system hardware level, the functionality of the hardware may be assigned in a manner that optimizes the system performance for a given load. At the processing engine level, loads may be balanced between the multiple processing modules of a given processing engine to further optimize the system performance.

[0099] Clusters of Systems

[0100] The systems described herein may also be clustered together in groups of two or more to provide additional processing power, storage connections, bandwidth, etc. Communication between two individual systems each configured similar to content delivery system **1010** may be made through network interface **1022** and/or **1023**. Thus, one content delivery system could communicate with another content delivery system through the network **1020** and/or **1024**. For example, a storage unit in one content delivery system could send data to a network interface engine of another content delivery system. As an example, these communications could be via TCP/IP protocols. Alternatively, the distributed interconnects **1080** of two content delivery systems **1010** may communicate directly. For example, a connection may be made directly between two switch fabrics, each switch fabric being the distributed interconnect **1080** of separate content delivery systems **1010**.

[0101] FIGS. 1G-1J illustrate four exemplary clusters of content delivery systems **1010**. It will be recognized that many other cluster arrangements may be utilized including more or less content delivery systems. As shown in FIGS. 1G-1J, each content delivery system may be configured as described above and include a distributive interconnect **1080** and a network interface processing engine **1030**. Interfaces **1022** may connect the systems to a network **1020**. As shown in **FIG. 1G**, two content delivery systems may be coupled together through the interface **1023** that is connected to each system's network interface processing engine **1030**. **FIG. 1H** shows three systems coupled together as in **FIG. 1G**. The interfaces **1023** of each system may be coupled directly together as shown, may be coupled together through a network or may be coupled through a distributed interconnect (for example a switch fabric).

[0102] **FIG. 1I** illustrates a cluster in which the distributed interconnects **1080** of two systems are directly coupled together through an interface **1500**. Interface **1500** may be any communication connection, such as a copper connection, optical fiber, wireless connection, etc. Thus, the distributed interconnects of two or more systems may directly communicate without communication through the processor engines of the content delivery systems **1010**. **FIG. 1J** illustrates the distributed interconnects of three systems directly communicating without first requiring communication through the processor engines of the content delivery systems **1010**. As shown in **FIG. 1J**, the interfaces **1500** each communicate with each other through another distributed interconnect **1600**. Distributed interconnect **1600** may be a switched fabric or any other distributed interconnect.

[0103] The clustering techniques described herein may also be implemented through the use of the management interface **1062**. Thus, communication between multiple content delivery systems **1010** also may be achieved through the management interface **1062**

[0104] Exemplary Data and Communication Flow Paths

[0105] **FIG. 1B** illustrates one exemplary data and communication flow path configuration among modules of one embodiment of content delivery system **1010**. The flow paths shown in **FIG. 1B** are just one example given to illustrate the significant improvements in data processing capacity and content delivery acceleration that may be realized using multiple content delivery engines that are individually optimized for different layers of the software stack and that are distributively interconnected as disclosed herein. The illustrated embodiment of **FIG. 1B** employs two network application processing modules **1070*a*** and **1070*b***, and two network transport processing modules **1050*a*** and **1050*b*** that are communicatively coupled with single storage management processing module **1040*a*** and single network interface processing module **1030*a***. The storage management processing module **1040*a*** is in turn coupled to content sources **1090** and **1100**. In **FIG. 1B**, inter-processor command or control flow (i.e. incoming or received data request) is represented by dashed lines, and delivered content data flow is represented by solid lines. Command and data flow between modules may be accomplished through the distributive interconnection **1080** (not shown), for example a switch fabric.

[0106] As shown in **FIG. 1B**, a request for content is received and processed by network interface processing module **1030*a*** and then passed on to either of network transport processing modules **1050*a*** or **1050*b*** for TCP/UDP processing, and then on to respective application processing modules **1070*a*** or **1070*b***, depending on the transport processing module initially selected. After processing by the appropriate network application processing module, the request is passed on to storage management processor **1040*a*** for processing and retrieval of the requested content from appropriate content sources **1090** and/or **1100**. Storage management processing module **1040*a*** then forwards the requested content directly to one of network transport processing modules **1050*a*** or **1050*b***, utilizing the capability of distributive interconnection **1080** to bypass network application processing modules **1070*a*** and **1070*b***. The requested content may then be transferred via the network interface processing module **1030*a*** to the external network **1020**. Benefits of bypassing the application processing modules with the delivered content include accelerated delivery of the requested content and offloading of workload from the application processing modules, each of which translate into greater processing efficiency and content delivery throughput. In this regard, throughput is generally measured in sustained data rates passed through the system and may be measured in bits per second. Capacity may be measured in terms of the number of files that may be partially cached, the number of TCP/IP connections per second as well as the number of concurrent TCP/IP connections that may be maintained or the number of simultaneous streams of a certain bit rate. In an alternative embodiment, the content may be delivered from the storage management processing module to the application processing module rather than bypassing the application processing module. This data flow may be advantageous if additional processing of the data is desired. For example, it may be desirable to decode or encode the data prior to delivery to the network.

[0107] To implement the desired command and content flow paths between multiple modules, each module may be provided with means for identification, such as a component ID. Components may be affiliated with content requests and content delivery to effect a desired module routing. The data-request generated by the network interface engine may include pertinent information such as the component ID of the various modules to be utilized in processing the request. For example, included in the data request sent to the storage management engine may be the component ID of the transport engine that is designated to receive the requested content data. When the storage management engine retrieves the data from the storage device and is ready to send the data to the next engine, the storage management engine knows which component ID to send the data to.

[0108] As further illustrated in **FIG. 1B**, the use of two network transport modules in conjunction with two network application processing modules provides two parallel processing paths for network transport and network application processing, allowing simultaneous processing of separate content requests and simultaneous delivery of separate content through the parallel processing paths, further increasing throughput/capacity and accelerating content delivery. Any two modules of a given engine may communicate with separate modules of another engine or may communicate with the same module of another engine. This is illustrated in **FIG. 1B** where the transport modules are shown to communicate with separate application modules and the

application modules are shown to communicate with the same storage management module.

[0109] **FIG. 1B** illustrates only one exemplary embodiment of module and processing flow path configurations that may be employed using the disclosed method and system. Besides the embodiment illustrated in **FIG. 1B**, it will be understood that multiple modules may be additionally or alternatively employed for one or more other network content delivery engines (e.g., storage management processing engine, network interface processing engine, system management processing engine, etc.) to create other additional or alternative parallel processing flow paths, and that any number of modules (e.g., greater than two) may be employed for a given processing engine or set of processing engines so as to achieve more than two parallel processing flow paths. For example, in other possible embodiments, two or more different network transport processing engines may pass content requests to the same application unit, or viceversa.

[0110] Thus, in addition to the processing flow paths illustrated in **FIG. 1B**, it will be understood that the disclosed distributive interconnection system may be employed to create other custom or optimized processing flow paths (e.g., by bypassing and/or interconnecting any given number of processing engines in desired sequence/s) to fit the requirements or desired operability of a given content delivery application. For example, the content flow path of **FIG. 1B** illustrates an exemplary application in which the content is contained in content sources **1090** and/or **1100** that are coupled to the storage processing engine **1040**. However as discussed above with reference to **FIG. 1A**, remote and/or live broadcast content may be provided to the content delivery system from the networks **1020** and/or **1024** via the second network interface connection **1023**. In such a situation the content may be received by the network interface engine **1030** over interface connection **1023** and immediately re-broadcast over interface connection **1022** to the network **1020**. Alternatively, content may be proceed through the network interface connection **1023** to the network transport engine **1050** prior to returning to the network interface engine **1030** for rebroadcast over interface connection **1022** to the network **1020** or **1024**. In yet another alternative, if the content requires some manner of application processing (for example encoded content that may need to be decoded), the content may proceed all the way to the application engine **1070** for processing. After application processing the content may then be delivered through the network transport engine **1050**, network interface engine **1030** to the network **1020** or **1024**.

[0111] In yet another embodiment, at least two network interface modules **1030*a*** and **1030*b*** may be provided, as illustrated in **FIG. 1A**. In this embodiment, a first network interface engine **1030*a*** may receive incoming data from a network and pass the data directly to the second network interface engine **1030*b*** for transport back out to the same or different network. For example, in the remote or live broadcast application described above, first network interface engine **1030*a*** may receive content, and second network interface engine **1030*b*** provide the m content to the network **1020** to fulfill requests from one or more clients for this content. Peer-to-peer level communication between the two network interface engines allows first network interface engine **1030*a*** to send the content directly to second network interface engine **1030*b*** via distributive interconnect **1080**. If necessary, the content may also be routed through transport processing engine **1050**, or through network transport processing engine **1050** and application go processing engine **1070**, in a manner described above.

[0112] Still yet other applications may exist in which the content required to be delivered is contained both in the attached content sources **1090** or **1100** and at other remote content sources. For example in a web caching application, not all content may be cached in the attached content sources, but rather some data may also be cached remotely. In such an application, the data and communication flow may be a combination of the various flows described above for content provided from the content sources **1090** and **1100** and for content provided from remote sources on the networks **1020** and/or **1024**.

[0113] The content delivery system **1010** described above is configured in a peer-to-peer manner that allows the various engines and modules to communicate with each other directly as peers through the distributed interconnect. This is contrasted with a traditional server architecture in which there is a main CPU. Furthermore unlike the arbitrated bus of traditional servers, the distributed interconnect **1080** provides a switching means which is not arbitrated and allows multiple simultaneous communications between the various peers. The data and communication flow may bypass unnecessary peers such as the return of data from the storage management processing engine **1060** directly to the network interface processing engine **1030** as described with reference to **FIG. 1B**.

[0114] Communications between the various processor engines may be made through the use of a standardized internal protocol. Thus, a standardized method is provided for routing through the switch fabric and communicating between any two of the processor engines which operate as peers in the peer to peer environment. The standardized internal protocol provides a mechanism upon which the external network protocols may "ride" upon or be incorporated within. In this manner additional internal protocol layers relating to internal communication and data exchange may be added to the external protocol layers. The additional internal layers may be provided in addition to the external layers or may replace some of the external protocol layers (for example as described above portions of the external headers may be replaced by identifiers or tags by the network interface engine).

[0115] The standardized internal protocol may consist of a system of message classes, or types, where the different classes can independently include fields or layers that are utilized to identify the destination processor engine or processor module for communication, control, or data messages provided to the switch fabric along with information pertinent to the corresponding message class. The standardized internal protocol may also include fields or layers that identify the priority that a data packet has within the content delivery system. These priority levels may be set by each processing engine based upon system-wide policies. Thus, some traffic within the content delivery system may be prioritized over other traffic and this priority level may be directly indicated within the internal protocol call scheme utilized to enable communications within the system. The prioritization helps enable the predictive traffic flow between

engines and end-to-end through the system such that service level guarantees may be supported.

[0116] Other internally added fields or layers may include processor engine state, system timestamps, specific message class identifiers for message routing across the switch fabric and at the receiving processor engine(s), system keys for secure control message exchange, flow control information to regulate control and data traffic flow and prevent congestion, and specific address tag fields that allow hardware at the receiving processor engines to move specific types of data directly into system memory.

[0117] In one embodiment, the internal protocol may be structured as a set, or system of messages with common system defined headers that allows all processor engines and, potentially, processor engine switch fabric attached hardware, to interpret and process messages all efficiently and intelligently. This type of design allows each processing engine, and specific functional entities within the processor engines, to have their own specific message classes optimized functionally for the exchanging their specific types control and data information. Some message classes that may be employed are: System Control messages for system management, Network Interface to Network Transport messages, Network Transport to Application Interface messages, File System to Storage engine messages, Storage engine to Network Transport messages, etc. Some of the fields of the standardized message header may include message priority, message class, message class identifier (subtype), message size, message options and qualifier fields, message context identifiers or tags, etc. In addition, the system statistics gathering, management and control of the various engines may be performed across the switch fabric connected system using the messaging capabilities.

[0118] By providing a standardized internal protocol, overall system performance may be improved. In particular, communication speed between the processor engines across the switch fabric may be increased. Further, communications between any two processor engines may be enabled. The standardized protocol may also be utilized to reduce the processing loads of a given engine by reducing the amount of data that may need to be processed by a given engine.

[0119] The internal protocol may also be optimized for a particular system application, providing further performance improvements. However, the standardized internal communication protocol may be general enough to support encapsulation of a wide range of networking and storage protocols. Further, while internal protocol may run on PCI, PCI-X, ATM, IB, Lightening I/O, the internal protocol is a protocol above these transport-level standards and is optimal for use in a switched (non-bus) environment such as a switch fabric. In addition, the internal protocol may be utilized to communicate devices (or peers) connected to the system in addition to those described herein. For example, a peer need not be a processing engine. In one example, a peer may be an ASIC protocol converter that is coupled to the distributed interconnect as a peer but operates as a slave device to other master devices within the system. The internal protocol may also be as a protocol communicated between systems such as used in the clusters described above.

[0120] Thus a system has been provided in which the networking/server clustering/storage networking has been collapsed into a single system utilizing a common low-overhead internal communication protocol/transport system.

[0121] Content Delivery Acceleration

[0122] As described above, a wide range of techniques have been provided for accelerating content delivery from the content delivery system **1010** to a network. By accelerating the speed at which content may be delivered, a more cost effective and higher performance system may be provided. These techniques may be utilized separately or in various combinations.

[0123] One content acceleration technique involves the use of a multi-engine system with dedicated engines for varying processor tasks. Each engine can perform operations independently and in parallel with the other engines without the other engines needing to freeze or halt operations. The engines do not have to compete for resources such as memory, I/O, processor time, etc. but are provided with their own resources. Each engine may also be tailored in hardware and/or software to perform specific content delivery task, thereby providing increasing content delivery speeds while requiring less system resources. Further, all data, regardless of the flow path, gets processed in a staged pipeline fashion such that each engine continues to process its layer of functionality after forwarding data to the next engine/layer.

[0124] Content acceleration is also obtained from the use of multiple processor modules within an engine. In this manner, parallelism may be achieved within a specific processing engine. Thus, multiple processors responding to different content requests may be operating in parallel within one engine.

[0125] Content acceleration is also provided by utilizing the multi-engine design in a peer to peer environment in which each engine may communicate as a peer. Thus, the communications and data paths may skip unnecessary engines. For example, data may be communicated directly from the storage processing engine to the transport processing engine without have to utilize resources of the application processing engine.

[0126] Acceleration of content delivery is also achieved by removing or stripping the contents of some protocol layers in one processing engine and replacing those layers with identifiers or tags for use with the next processor engine in the data or communications flow path. Thus, the processing burden placed on the subsequent engine may be reduced. In addition, the packet size transmitted across the distributed interconnect may be reduced. Moreover, protocol processing may be off-loaded from the storage and/or application processors, thus freeing those resources to focus on storage or application processing.

[0127] Content acceleration is also provided by using network processors in a network endpoint system. Network processors generally are specialized to perform packet analysis functions at intermediate network nodes, but in the content delivery system disclosed the network processors have been adapted for endpoint functions. Furthermore, the parallel processor configurations within a network processor allow these endpoint functions to be performed efficiently.

[0128] In addition, content acceleration has been provided through the use of a distributed interconnection such as a switch fabric. A switch fabric allows for parallel communications between the various engines and helps to efficiently implement some of the acceleration techniques described herein.

[0129] It will be recognized that other aspects of the content delivery system **1010** also provide for accelerated delivery of content to a network connection. Further, it will be recognized that the techniques disclosed herein may be equally applicable to other network endpoint systems and even non-endpoint systems.

[0130] Exemplary Hardware Embodiments

[0131] FIGS. 1C-1F illustrate just a few of the many multiple network content delivery engine configurations possible with one exemplary hardware embodiment of content delivery system **1010**. In each illustrated configuration of this hardware embodiment, content delivery system **1010** includes processing modules that may be configured to operate as content delivery engines **1030**, **1040**, **1050**, **1060**, and **1070** communicatively coupled via distributive interconnection **1080**. As shown in **FIG. 1C**, a single processor module may operate as the network interface processing engine **1030** and a single processor module may operate as the system management processing engine **1060**. Four processor modules **1001** may be configured to operate as either the transport processing engine **1050** or the application processing engine **1070**. Two processor modules **1003** may operate as either the storage processing engine **1040** or the transport processing engine **1050**. The Gigabit (Gb) Ethernet front end interface **1022**, system management interface **1062** and dual fibre channel arbitrated loop **1092** are also shown.

[0132] As mentioned above, the distributive interconnect **1080** may be a switch fabric based interconnect. As shown in **FIG. 1C**, the interconnect may be an IBM PRIZMA-E eight/sixteen port switch fabric **1081**. In an eight port mode, this switch fabric is an 8×3.54 Gbps fabric and in a sixteen port mode, this switch fabric is a 16×1.77 Gbps fabric. The eight/sixteen port switch fabric may be utilized in an eight port mode for performance optimization. The switch fabric **1081** may be coupled to the individual processor modules through interface converter circuits **1082**, such as IBM UDASL switch interface circuits. The interface converter circuits **1082** convert the data aligned serial link interface (DASL) to a UTOPIA (Universal Test and Operations PHY Interface for ATM) parallel interface. FPGAs (field programmable gate array) may be utilized in the processor modules as a fabric interface on the processor modules as shown in **FIG. 1C**. These fabric interfaces provide a 64/66 Mhz PCI interface to the interface converter circuits **1082**. **FIG. 4** illustrates a functional block diagram of such a fabric interface **34**. As explained below, the interface **34** provides an interface between the processor module bus and the UDASL switch interface converter circuit **1082**. As shown in **FIG. 4**, at the switch fabric side, a physical connection interface **41** provides connectivity at the physical level to the switch fabric. An example of interface **41** is a parallel bus interface complying with the UTOPIA standard. In the example of **FIG. 4**, interface **41** is a UTOPIA 3 interface providing a 32-bit 110 Mhz connection. However, the concepts disclosed herein are not protocol dependent and the switch fabric need not comply with any particular ATM or non ATM standard.

[0133] Still referring to **FIG. 4**, SAR (segmentation and reassembly) unit **42** has appropriate SAR logic **42***a* for performing segmentation and reassembly tasks for converting messages to fabric cells and vice-versa as well as message classification and message class-to-queue routing, using memory **42***b* and **42***c* for transmit and receive queues. This permits different classes of messages and permits the classes to have different priority. For example, control messages can be classified separately from data messages, and given a different priority. All fabric cells and the associated messages may be self routing, and no out of band signaling is required.

[0134] A special memory modification scheme permits one processor module to write directly into memory of another. This feature is facilitated by switch fabric interface **34** and in particular by its message classification capability. Commands and messages follow the same path through switch fabric interface **34**, but can be differentiated from other control and data messages. In this manner, processes executing on processor modules can communicate directly using their own memory spaces.

[0135] Bus interface **43** permits switch fabric interface **34** to communicate with the processor of the processor module via the module device or I/O bus. An example of a suitable bus architecture is a PCI architecture, but other architectures could be used. Bus interface **43** is a master/target device, permitting interface **43** to write and be written to and providing appropriate bus control. The logic circuitry within interface **43** implements a state machine that provides the communications protocol, as well as logic for configuration and parity.

[0136] Referring again to **FIG. 1C**, network processor **1032** (for example a MOTOROLA C-Port C-5 network processor) of the network interface processing engine **1030** may be coupled directly to an interface converter circuit **1082** as shown. As mentioned above and further shown in **FIG. 1C**, the network processor **1032** also may be coupled to the network **1020** by using a VITESSE GbE SERDES (serializer-deserializer) device (for example the VSC7123) and an SFP (small form factor pluggable) optical transceiver for LC fibre connection.

[0137] The processor modules **1003** include a fibre channel (FC) controller as mentioned above and further shown in **FIG. 1C**. For example, the fibre channel controller may be the LSI SYMFC929 dual 2 GBaud fibre channel controller. The fibre channel controller enables communication with the fibre channel **1092** when the processor module **1003** is utilized as a storage processing engine **1040**. Also illustrated in FIGS. 1C-1F is optional adjunct processing unit **1300** that employs a POWER PC processor with SDRAM. The adjunct processing unit is shown coupled to network processor **1032** of network interface processing engine **1030** by a PCI interface. Adjunct processing unit **1300** may be employed for monitoring system parameters such as temperature, fan operation, system health, etc.

[0138] As shown in FIGS. 1C-1F, each processor module of content delivery engines **1030**, **1040**, **1050**, **1060**, and **1070** is provided with its own synchronous dynamic random access memory ("SDRAM") resources, enhancing the independent operating capabilities of each module. The memory resources may be operated as ECC (error correcting code) memory. Network interface processing engine **1030** is also provided with static random access memory ("SRAM"). Additional memory circuits may also be utilized as will be recognized by those skilled in the art. For example, additional memory resources (such as synchronous SRAM and non-volatile FLASH and EEPROM) may be provided in

conjunction with the fibre channel controllers. In addition, boot FLASH memory may also be provided on the of the processor modules. The processor modules **1001** and **1003** of **FIG. 1C** may be configured in alternative manners to implement the content delivery processing engines such as the network interface processing engine **1030**, storage processing engine **1040**, transport processing engine **1050**, system management processing engine **1060**, and application processing engine **1070**. Exemplary configurations are shown in FIGS. **1D**-1F, however, it will be recognized that other configurations may be utilized.

[0139] As shown in **FIG. 1D**, two Pentium III based processing modules may be utilized as network application processing modules **1070***a* and **1070***b* of network application processing engine **1070**. The remaining two Pentium III-based processing modules are shown in **FIG. 1D** configured as network transport/protocol processing modules **1050***a* and **1050***b* of network transport/protocol processing engine **1050**. The embodiment of **FIG. 1D** also includes two POWER PC-based processor modules, configured as storage management processing modules **1040***a* and **1040***b* of storage management processing engine **1040**. A single MOTOROLA C-Port C-5 based network processor is shown employed as network interface processing engine **1030**, and a single Pentium III-based processing module is shown employed as system management processing engine **1060**.

[0140] In **FIG. 1E**, the same hardware embodiment of **FIG. 1C** is shown alternatively configured so that three Pentium III-based processing modules function as network application processing modules **1070***a*, **1070***b* and **1070***c* of network application processing engine **1070**, and so that the sole remaining Pentium III-based processing module is configured as a network transport processing module **1050***a* of network transport processing engine **1050**. As shown, the remaining processing modules are configured as in **FIG. 1D**.

[0141] In **FIG. 1F**, the same hardware embodiment of **FIG. 1C** is shown in yet another alternate configuration so that three Pentium III-based processing modules function as application processing modules **1070***a*, **1070***b* and **1070***c* of network application processing engine **1070**. In addition, the network transport processing engine **1050** includes one Pentium III-based processing module that is configured as network transport processing module **1050***a*, and one POWER PC-based processing module that is configured as network transport processing module **1050***b*. The remaining POWER PC-based processor module is configured as storage management processing module **1040***a* of storage management processing engine **1040**.

[0142] It will be understood with benefit of this disclosure that the hardware embodiment and multiple engine configurations thereof illustrated in FIGS. 1C-1F are exemplary only, and that other hardware embodiments and engine configurations thereof are also possible. It will further be understood that in addition to changing the assignments of individual processing modules to particular processing engines, distributive interconnect **1080** enables the vary processing flow paths between individual modules employed in a particular engine configuration in a manner as described in relation to **FIG. 1B**. Thus, for any given hardware embodiment and processing engine configuration, a number of different processing flow paths may be employed so as to optimize system performance to suit the needs of particular system applications.

[0143] Single Chassis Design

[0144] As mentioned above, the content delivery system **1010** may be implemented within a single chassis, such as for example, a 2 U chassis. The system may be expanded further while still remaining a single chassis system. In particular, utilizing a multiple processor module or blade arrangement connected through a distributive interconnect (for example a switch fabric) provides a system that is easily scalable. The chassis and interconnect may be configured with expansion slots provided for adding additional processor modules. Additional processor modules may be provided to implement additional applications within the same chassis. Alternatively, additional processor modules may be provided to scale the bandwidth of the network connection. Thus, though describe with respect to a 1 Gbps Ethernet connection to the external network, a 10 Gbps, 40 Gbps or more connection may be established by the system through the use of more network interface modules. Further, additional processor modules may be added to address a system's particular bottlenecks without having to expand all engines of the system. The additional modules may be added during a systems initial configuration, as an upgrade during system maintenance or even hot plugged during system operation.

[0145] Alternative Systems Configurations

[0146] Further, the network endpoint system techniques disclosed herein may be implemented in a variety of alternative configurations that incorporate some, but not necessarily all, of the concepts disclosed herein. For example, **FIGS. 2 and 2A** disclose two exemplary alternative configurations. It will be recognized, however, that many other alternative configurations may be utilized while still gaining the benefits of the inventions disclosed herein.

[0147] **FIG. 2** is a more generalized and functional representation of a content delivery system showing how such a system may be alternately configured to have one or more of the features of the content delivery system embodiments illustrated in FIGS. **1A-1F**. **FIG. 2** shows content delivery system **200** coupled to network **260** from which content requests are received and to which content is delivered. Content sources **265** are shown coupled to content delivery system **200** via a content delivery flow path **263** that may be, for example, a storage area network that links multiple content sources **265**. A flow path **203** may be provided to network connection **272**, for example, to couple content delivery system **200** with other network appliances, in this case one or more servers **201** as illustrated in **FIG. 2**.

[0148] In **FIG. 2** content delivery system **200** is configured with multiple processing and memory modules that are distributively interconnected by inter-process communications path **230** and inter-process data movement path **235**. Inter-process communications path **230** is provided for receiving and distributing inter-processor command communications between the modules and network **260**, and interprocess data movement path **235** is provided for receiving and distributing inter-processor data among the separate modules. As illustrated in FIGS. **1A-1F**, the functions of inter-process communications path **230** and inter-process data movement path **235** may be together handled by a single distributive interconnect **1080** (such as a switch fabric, for example), however, it is also possible to separate the communications and data paths as illustrated in **FIG. 2**, for example using other interconnect technology.

[0149] FIG. 2 illustrates a single networking subsystem processor module **205** that is provided to perform the combined functions of network interface processing engine **1030** and transport processing engine **1040** of FIG. 1A. Communication and content delivery between network **260** and networking subsystem processor module **205** are made through network connection **270**. For certain applications, the functions of network interface processing engine **1030** and transport processing engine **1050** of FIG. 1A may be so combined into a single module **205** of FIG. 2 in order to reduce the level of communication and data traffic handled by communications path **230** and data movement path **235** (or single switch fabric), without adversely impacting the resources of application processing engine or subsystem module. If such a modification were made to the system of FIG. 1A, content requests may be passed directly from the combined interface/transport engine to network application processing engine **1070** via distributive interconnect **1080**. Thus, as previously described the functions of two or more separate content delivery system engines may be combined as desired (e.g., in a single module or in multiple modules of a single processing blade), for example, to achieve advantages in efficiency or cost.

[0150] In the embodiment of FIG. 2, the function of network application processing engine **1070** of FIG. 1A is performed by application processing subsystem module **225** of FIG. 2 in conjunction with application RAM subsystem module **220** of FIG. 2. System monitor module **240** communicates with server/s **201** through flow path **203** and Gb Ethernet network interface connection **272** as also shown in FIG. 2. The system monitor module **240** may provide the function of the system management engine **1060** of FIG. 1A and/or other system policy/filter functions such as may also be implemented in the network interface processing engine **1030** as described above with reference to FIG. 1A.

[0151] Similarly, the function of network storage management engine **1040** is performed by storage subsystem module **210** in conjunction with file system cache subsystem module **215**. Communication and content delivery between content sources **265** and storage subsystem module **210** are shown made directly through content delivery flowpath **263** through fibre channel interface connection **212**. Shared resources subsystem module **255** is shown provided for access by each of the other subsystem modules and may include, for example, additional processing resources, additional memory resources such as RAM, etc.

[0152] Additional processing engine capability (e.g., additional system management processing capability, additional application processing capability, additional storage processing capability, encryption/decryption processing capability, compression/decompression processing capability, encoding/decoding capability, other processing capability, etc.) may be provided as desired and is represented by other subsystem module **275**. Thus, as previously described the functions of a single network processing engine may be sub-divided between separate modules that are distributively interconnected. The sub-division of network processing engine tasks may also be made for reasons of efficiency or cost, and/or may be taken advantage of to allow resources (e.g., memory or processing) to be shared among separate modules. Further, additional shared resources may be made available to one or more separate modules as desired.

[0153] Also illustrated in FIG. 2 are optional monitoring agents **245** and resources **250**. In the embodiment of FIG. 2, each monitoring agent **245** may be provided to monitor the resources **250** of its respective processing subsystem module, and may track utilization of these resources both within the overall system **200** and within its respective processing subsystem module. Examples of resources that may be so monitored and tracked include, but are not limited to, processing engine bandwidth, Fibre Channel bandwidth, number of available drives, IOPS (input/output operations per second) per drive and RAID (redundant array of inexpensive discs) levels of storage devices, memory available for caching blocks of data, table lookup engine bandwidth, availability of RAM for connection control structures and outbound network bandwidth availability, shared resources (such as RAM) used by streaming application on a per-stream basis as well as for use with connection control structures and buffers, bandwidth available for message passing between subsystems, bandwidth available for passing data between the various subsystems, etc.

[0154] Information gathered by monitoring agents **245** may be employed for a wide variety of purposes including for billing of individual content suppliers and/or users for pro-rata use of one or more resources, resource use analysis and optimization, resource health alarms, etc. In addition, monitoring agents may be employed to enable the deterministic delivery of content by system **200** as described in concurrently filed, co-pending United States patent application No. _____, entitled "System and Method for the Deterministic Delivery of Data and Services," which is incorporated herein by reference.

[0155] In operation, content delivery system **200** of FIG. 2 may be configured to wait for a request for content or services prior to initiating content delivery or performing a service. A request for content, such as a request for access to data, may include, for example, a request to start a video stream, a request for stored data, etc. A request for services may include, for example, a request for to run an application, to store a file, etc. A request for content or services may be received from a variety of sources. For example, if content delivery system **200** is employed as a stream server, a request for content may be received from a client system attached to a computer network or communication network such as the Internet. In a larger system environment, e.g., a data center, a request for content or services may be received from a separate subcomponent or a system management processing engine, that is responsible for performance of the overall system or from a sub-component that is unable to process the current request. Similarly, a request for content or services may be received by a variety of components of the receiving system. For example, if the receiving system is a stream server, networking subsystem processor module **205** might receive a content request. Alternatively, if the receiving system is a component of a larger system, e.g., a data center, system management processing engine may be employed to receive the request.

[0156] Upon receipt of a request for content or services, the request may be filtered by system monitor **240**. Such filtering may serve as a screening agent to filter out requests that the receiving system is not capable of processing (e.g., requests for file writes from read-only system embodiments, unsupported protocols, content/services unavailable on system **200**, etc.). Such requests may be rejected outright and

the requestor notified, may be re-directed to a server **201** or other content delivery system **200** capable of handling the request, or may be disposed of any other desired manner.

[0157] Referring now in more detail to one embodiment of **FIG. 2** as may be employed in a stream server configuration, networking processing subsystem module **205** may include the hardware and/or software used to run TCP/IP (Transmission Control Protocol/Internet Protocol), UDP/IP (User Datagram Protocol/Internet Protocol), RTP (Real-Time Transport Protocol), Internet Protocol (IP), Wireless Application Protocol (WAP) as well as other networking protocols. Network interface connections **270** and **272** may be considered part of networking subsystem processing module **205** or as separate components. Storage subsystem module **210** may include hardware and/or software for running the Fibre Channel (FC) protocol, the SCSI (Small Computer Systems Interface) protocol, iSCSI protocol as well as other storage networking protocols. FC interface **212** to content delivery flowpath **263** may be considered part of storage subsystem module **210** or as a separate component. File system cache subsystem module **215** may include, in addition to cache hardware, one or more cache management algorithms as well as other software routines.

[0158] Application RAM subsystem module **220** may function as a memory allocation subsystem and application processing subsystem module **225** may function as a stream-serving application processor bandwidth subsystem. Among other services, application RAM subsystem module **220** and application processing subsystem module **225** may be used to facilitate such services as the pulling of content from storage and/or cache, the formatting of content into RTSP (Real-Time Streaming Protocol) or another streaming protocol as well the passing of the formatted content to networking subsystem **205**.

[0159] As previously described, system monitor module **240** may be included in content delivery system **200** to manage one or more of the subsystem processing modules, and may also be used to facilitate communication between the modules.

[0160] In part to allow communications between the various subsystem modules of content delivery system **200**, inter-process communication path **230** may be included in content delivery system **200**, and may be provided with its own monitoring agent **245**. Inter-process communications path **230** may be a reliable protocol path employing a reliable IPC (Interprocess Communications) protocol. To allow data or information to be passed between the various subsystem modules of content delivery system **200**, inter-process data movement path **235** may also be included in content delivery system **200**, and may be provided with its own monitoring agent **245**. As previously described, the functions of inter-process communications path **230** and inter-process data movement path **235** may be together handled by a single distributive interconnect **1080**, that may be a switch fabric configured to support the bandwidth of content being served.

[0161] In one embodiment, access to content source **265** may be provided via a content delivery flow path **263** that is a fibre channel storage area network (SAN), a switched technology. In addition, network connectivity may be provided at network connection **270** (e.g., to a front end network) and/or at network connection **272** (e.g., to a back end network) via switched gigabit Ethernet in conjunction with the switch fabric internal communication system of content delivery system **200**. As such, that the architecture illustrated in **FIG. 2** may be generally characterized as equivalent to a networking system.

[0162] One or more shared resources subsystem modules **255** may also be included in a stream server embodiment of content delivery system **200**, for sharing by one or more of the other subsystem modules. Shared resources subsystem module **255** may be monitored by the monitoring agents **245** of each subsystem sharing the resources. The monitoring agents **245** of each subsystem module may also be capable of tracking usage of shared resources **255**. As previously described, shared resources may include RAM (Random Access Memory) as well as other types of shared resources.

[0163] Each monitoring agent **245** may be present to monitor one or more of the resources **250** of its subsystem processing module as well as the utilization of those resources both within the overall system and within the respective subsystem processing module. For example, monitoring agent **245** of storage subsystem module **210** may be configured to monitor and track usage of such resources as processing engine bandwidth, Fibre Channel bandwidth to content delivery flow path **263**, number of storage drives attached, number of input/output operations per second (IOPS) per drive and RAID levels of storage devices that may be employed as content sources **265**. Monitoring agent **245** of file system cache subsystem module **215** may be employed monitor and track usage of such resources as processing engine bandwidth and memory employed for caching blocks of data. Monitoring agent **245** of networking subsystem processing module **205** may be employed to monitor and track usage of such resources as processing engine bandwidth, table lookup engine bandwidth, RAM employed for connection control structures and outbound network bandwidth availability. Monitoring agent **245** of application processing subsystem module **225** may be employed to monitor and track usage of processing engine bandwidth. Monitoring agent **245** of application RAM subsystem module **220** may be employed to monitor and track usage of shared resource **255**, such as RAM, which may be employed by a streaming application on a per-stream basis as well as for use with connection control structures and buffers. Monitoring agent **245** of inter-process communication path **230** may be employed to monitor and track usage of such resources as the bandwidth used for message passing between subsystems while monitoring agent **245** of inter-process data movement path **235** may be employed to monitor and track usage of bandwidth employed for passing data between the various subsystem modules.

[0164] The discussion concerning **FIG. 2** above has generally been oriented towards a system designed to deliver streaming content to a network such as the Internet using, for example, Real Networks, Quick Time or Microsoft Windows Media streaming formats. However, the disclosed systems and methods may be deployed in any other type of system operable to deliver content, for example, in web serving or file serving system environments. In such environments, the principles may generally remain the same. However for application processing embodiments, some differences may exist in the protocols used to communicate and the method by which data delivery is metered (via streaming protocol, versus TCP/IP windowing).

[0165] FIG. 2A illustrates an even more generalized network endpoint computing system that may incorporate at least some of the concepts disclosed herein. As shown in FIG. 2A, a network endpoint system 10 may be coupled to an external network 11. The external network 11 may include a network switch or router coupled to the front end of the endpoint system 10. The endpoint system 10 may be alternatively coupled to some other intermediate network node of the external network. The system 10 may further include a network engine 9 coupled to an interconnect medium 14. The network engine 9 may include one or more network processors. The interconnect medium 14 may be coupled to a plurality of processor units 13 through interfaces 13*a*. Each processor unit 13 may optionally be couple to data storage (in the exemplary embodiment shown each unit is couple to data storage). More or less processor units 13 may be utilized than shown in FIG. 2A.

[0166] The network engine 9 may be a processor engine that performs all protocol stack processing in a single processor module or alternatively may be two processor modules (such as the network interface engine 1030 and transport engine 1050 described above) in which split protocol stack processing techniques are utilized. Thus, the functionality and benefits of the content delivery system 1010 described above may be obtained with the system 10. The interconnect medium 14 may be a distributive interconnection (for example a switch fabric) as described with reference to FIG. 1A. All of the various computing, processing, communication, and control techniques described above with reference to FIGS. 1A-1F and 2 may be implemented within the system 10. It will therefore be recognized that these techniques may be utilized with a wide variety of hardware and computing systems and the techniques are not limited to the particular embodiments disclosed herein.

[0167] The system 10 may consist of a variety of hardware configurations. In one configuration the network engine 9 may be a stand-alone device and each processing unit 13 may be a separate server. In another configuration the network engine 9 may be configured within the same chassis Who as the processing units 13 and each processing unit 13 may be a separate server card or other computing system. Thus, a network engine (for example an engine containing a network processor) may provide transport acceleration and be combined with multi-server functionality within the system 10. The system 10 may also include shared management and interface components. Alternatively, each processing unit 13 may be a processing engine such as the transport processing engine, application engine, storage engine, or system management engine of FIG. 1A. In yet another alternative, each processing unit may be a processor module (or processing blade) of the processor engines shown in the system of FIG. 1A.

[0168] FIG. 2B illustrates yet another use of a network engine 9. As shown in FIG. 2B, a network engine 9 may be added to a network interface card 35. The network interface card 35 may further include the interconnect medium 14 which may be similar to the distributed interconnect 1080 described above. The network interface card may be part of a larger computing system such as a server. The network interface card may couple to the larger system through the interconnect medium 14. In addition to the functions described above, the network engine 9 may perform all traditional functions of a network interface card.

[0169] It will be recognized that all the systems described above (FIGS. 1A, 2, 2A, and 2B) utilize a network engine between the external network and the other processor units that are appropriate for the function of the particular network node. The network engine may therefore offload tasks from the other processors. The network engine also may perform "look ahead processing" by performing processing on a request before the request reaches whatever processor is to perform whatever processing is appropriate for the network node. In this manner, the system operations may be accelerated and resources utilized more efficiently.

[0170] Network Security

[0171] The computing systems disclosed herein may incorporate the use of security measures to protect against network security attacks, including but not limited to, DoS attacks, syn attacks, ping attacks, Trojan horse attacks, unauthorized access attacks, etc. The security measures disclosed herein may be incorporated within a network endpoint computing system. More particularly, the security measures may be implemented with intelligent security hardware that is placed at the network interface of the endpoint system. The intelligent security hardware may be coupled on one side to the external network and on the other side to the rest of the endpoint computing system through an interconnect medium. The interconnection medium may be in one example a switch fabric. As used herein intelligent security hardware may also be called a security accelerator.

[0172] Preferably the intelligent security hardware is programmable and has the capability of performing "look ahead processing" so that processing may be performed on network packets in order to reduce the workload placed upon the endpoint systems resources. For example, the intelligent security hardware may include the capability of decoding incoming network packet headers and deciding where to forward the packet. One such type of intelligent hardware is a network processor or a network processor coupled with other processors or hardware. Any hardware capable of performing the functions described herein, however, may be suitable. The intelligent security hardware may receive incoming packets, decode headers, and determine if the packets are authentic or if the packets are part of a security attack. The intelligent security hardware may be programmable to implement any of a wide variety of security algorithms utilized to detect an attack or unauthorized access. Thus although the security functions are hardware based, the intelligent security hardware offers flexibility in being configured for serving a variety of security functions and for addressing a variety of changing security threats.

[0173] By providing intelligent security hardware (or a security accelerator) that can perform security functions at the front end of the endpoint computing system, the often complex and CPU intensive security functions may be off-loaded from the rest of the endpoint computing system. Security protections are, therefore, implemented without impacting the performance of the rest of the endpoint computing system. Further, unlike a conventional network front end, such as the network controllers of conventional servers, and unlike many firewalls, the intelligent security hardware need not DMA (direct memory access) packets into RAM. Instead, it processes the packets as they arrive. This processing is sometimes referred to as "pass through processing". Thus, attacks may be detected quickly without

inundating resources at the endpoint. In this manner, the security functions may be accelerated. The intelligent security hardware described herein may be considered a security accelerator.

[0174] Thus, the intelligent security hardware (or security accelerator) may be utilized in network endpoint system connected to a network that carries data in packet format. A security accelerator is programmed to receive packets from the network and to examine each packet to determine whether data in the packet represents a potential security violation. The network endpoint system may further include one or more processing units that are programmed to execute some form of server or client application programming or content delivery and to thereby respond to requests contained within the packets. An interconnection medium may be used to directly connect the security accelerator to the processing units.

[0175] The security accelerator may be utilized at the front end of the endpoint, and thus, eliminate the need for a firewall. Security tools are offloaded from other endpoint resources to the security accelerator, so that the other resources can be devoted to the other tasks that the endpoint system is to perform. The security accelerator's "look ahead" processing unburdens the processing units that must perform the basic tasks appropriate for that particular network endpoint.

[0176] The security accelerator may be implemented with either a network processor or a CPU type general processor. When a network processor is used, it performs "pass through" type processing that especially suitable for many types of security algorithms. This type of processing can be more suited for these algorithms than the state-modification intensive and memory-access intensive processing of a CPU type processor.

[0177] The security accelerator can detect attempted security breaches very quickly. It can take immediate action, such as discarding the packet or notifying the network administrator. By utilizing a programmable security accelerator, the security accelerator may be easily updated by means of a simple download, making it easy to upgrade the security accelerator to detect new types of attacks. Thus, the security accelerator provides flexibility for providing counter measures to new security attacks as the new types of attacks become known.

[0178] As mentioned above, the intelligent security hardware or security accelerator may be implemented through the use of a network processor. The use of a network processor for implementing the security functions will initially be discussed for illustrative purposes with regards to the content delivery systems of FIGS. **1A-1F** and **2** and the network endpoint computing system of **FIG. 2A**, all of which are discussed above. It will be recognized that the security techniques described may be utilized in other systems or applications as will be described in more detail below.

[0179] As described above, the content delivery system **1010** of FIGS. 1A-1F may include a network interface processing engine **1030** which may include one or more network interface modules The network interface modules may include a network processor. The network interface processing engine operates as an interface between the external network and the distributed interconnection **1080** (for example a switch fabric). Similarly, the networking subsystem module of **FIG. 2** may include a network processor and the system of **FIG. 2A** may include a network processor **12**. As described above, a network processor is generally used in network switching devices and is specialized in decoding incoming network packet headers and deciding where to forward the packet. The network processor in all of theses systems may be programmed to provide the security measures described herein to achieve the benefits mentioned above. the specialized abilities of a network processor are therefore being applied security measures, a non-standard function for network processors.

[0180] Exemplary Security Measures

[0181] A variety of exemplary security measures are discussed herein with reference to utilizing a network processor as the intelligent security hardware/security accelerator. It will be recognized that other hardware may be utilized to achieve the benefits described herein.

[0182] As mentioned above, a network processor can be programmed to implement a fill array of security or firewall tools. Though some security measures are described below, it will be recognized that the programmability of a network processor will allow any number of security measures to be implemented.

[0183] A common security attack is a denial of service (DoS) attack. To protect against DoS attacks, the network processor is programmed to receive incoming packets, decode headers, and determine whether the packet is authentic. If the packet is authenticated, the network processor forwards it to the appropriate processor module or unit or other sub-system of the computing system. The techniques programmed into the network processor that determines whether a packet is part of a DoS attack may be implemented with various new and/or known techniques. For example, the receipt of a large number of new TCP open connection requests over a very short time period from the same source is indicative of a DoS attack. As is known in the field of network security, various tables, counters, and logic may be used to determine if a DoS attack is occurring.

[0184] Unlike a conventional network front end, such as the network controllers of conventional servers, and unlike many firewalls, the network processor does not DMA (direct memory access) packets into RAM. Instead, it processes the packets as they arrive. This processing is sometimes referred to as "pass through processing". Thus, DoS attacks may be detected quickly without inundating resources at the endpoint.

[0185] A "syn" attack is another potential network attack. In a syn attack, syn requests cause the TCP/IP stack to overfill. To counter this type of attack, the network processor may be programmed to monitor the frequency and same-source syn requests. If a threshold number of either of these counts is exceeded, the network processor is programmed to take action, such as throttling down requests or alerting a network administrator.

[0186] Other types of attacks make use of "bogus" source addresses. To thwart such attacks, the network processor can be programmed with various algorithms directed to authenticating source IP addresses. Yet another attack is known as "ping" attacks. In ping attacks, a ping request is received at

one system or network (the "amplifier") with a broadcast destination request. The node that is the victim of the attack is falsely listed as the address to which ping responses will be directed. The amplifier that received the ping request will broadcast the request to many endpoints within its network, thus generating massive amounts of ping replies directed at the victim of the attack. To thwart this attack at the victim endpoint, a network processor at the victim's endpoint system may be programmed to detect a massive amount of replies to a ping. To thwart this attack at the amplifier receiving the ping request, a network processor in the amplifier may be programmed to ignore broadcast ping requests or to limit the number of ping responses provided from the network.

[0187] The network processor may be further programmed to implement authentication verification and access control lists (ACL). This type of security task is thereby offloaded from the other processing or CPU resources. For example, an authentication handshake that verifies the identity of a user could be offloaded from the other resources of the network endpoint system. Furthermore, once the user is identified, the network processor may be programmed to determine what content is accessible by that user and discard unauthorized content access requests.

[0188] The network processor can be further programmed to implement packet filtering. In packet filtering, the header information of a packet may be analyzed and compared to large filter lists. A common limitation of filtering in traditional security systems is that large filter lists are required, leading to excessive memory and processor bandwidth usage to search for a filter match for every incoming packet. By providing a network processor separate from the other processor resources, the filter lists are not limited by other system resources such as memory and CPU bandwidth. Further complex filters used to analyze many different header types may be implemented without impacting the other processing resources.

[0189] A Trojan horse attack is another security attack. In a Trojan horse attack a program may execute to cause an unauthorized data transfer out to the external network. A firewall may assume that the data being transferred out was from an authorized user, whereas the network processor could be programmed to permit outgoing data to be sent only to authorized users. Alternatively, the network processor may be programmed to allow outgoing data to only be initiated by trusted connections. Thus, the network processor may not only provide security related to incoming events, but also provide security related to outgoing events. In this sense, it network processor provides "bi-directional" security.

[0190] The use of intelligent security hardware such as the programmable network processor further allows security provisions to be especially tailored to the type of network computing system that incorporates the hardware. For example, the network endpoint systems of FIGS. **1A-1F** and **2** are content delivery systems. In these systems the network processor permits the security tools to be programmed especially for content delivery systems. The network processor could be programmed to allow access only for types of requests that are supported by the content delivery system. Thus, if the content delivery system only serves HTTP requests, the security tools can be directed specifically to that protocol and known security weaknesses in that protocols. The network processor could also be further programmed to discard all non HTTP requests. Alternatively, if a system is known not to support a specific protocol, then the network processor may be programmed to disregard all connection requests for that type of protocol. A system that does not serve FTP connections may be, for example, programmed to discard all FTP connection requests.

[0191] Therefore, the security measures described herein may provide full firewall functionality and eliminate the need for a separate firewall device. Although many of the exemplary security measures have been described with reference to a network processor, it will be recognized that other programmable hardware may implement this same functionality.

[0192] Other Systems for Using the Securty Hardware

[0193] The exemplary security techniques described above have been primarily described with reference to a network endpoint system such as a content delivery system configured in a multi-module (or multi-blade) manner. However, the security techniques are not limited to use in such systems and many alternative network endpoint systems may utilize these techniques. Further, although generally described with relation to a server type endpoint, a client endpoint system could also have intelligent security hardware (such as a network processor) programmed to implement security tools for attacks that might occur at a client system.

[0194] In the examples of FIGS. **5-8** described below, the network processing systems are network endpoint systems. The security techniques described are not, however, limited to only network endpoints but may also be utilized within hardware located at various intermediate network nodes. The hardware located at various intermediate nodes that incorporates the security techniques described herein may include routers, switches or other network hardware. In general, the intelligent security hardware (or security accelerator) could be at any network node. Although the various systems differ in their overall architectures, in each system, the security accelerator executes security tools of the type described above.

[0195] No matter what type of system is utilized, a common characteristic of each system is that the security accelerator resides between a network and other processing units that are appropriate for a given network node or between a network and other network nodes. The security accelerator may be, but is not necessarily implemented with a network processor. A CPU type general purpose processor may be used instead of, or in addition to a network processor. Or, either a network processor or a CPU could be used with additional hardware logic. Regardless of what security hardware is used to implement the security accelerator, the security hardware performs "look ahead"processing on data as it is received. This processing, specifically directed to executing network security tools, is performed on data before the data reaches whatever device is to perform whatever basic processing is appropriate for the network node or before the data is routed to the next network node. The security accelerator thereby offloads the security processing from those processing units or network nodes.

[0196] **FIG. 5** illustrates a system **50** in which security accelerator **53** is a stand alone unit that is a separate physical

entity from servers **51**. The security accelerator **53** and the servers **51** are connected by an interconnection medium **52**. The security accelerator **53** has an Ethernet connection on the front end and an interconnection medium connection on the back end. The interconnection medium **52** could be any message passing medium, including those described above, e.g., switch fabric, bus, or shared memory. Alternatively, a network connection such as a LAN, could be used.

[0197] **FIG. 6** illustrates a multi-slot chassis or fixed configuration chassis system **60**. In system **60**, the security accelerator **63** and servers **61** are implemented as cards within the same physical chassis, connected by an interconnection medium **62**. Interconnection medium **62** may be any of the various interconnection media described above.

[0198] **FIG. 7** illustrates a security accelerator **75** in a system **70** that is similar to the system **60** except that the functionality of the server cards **71-73** has been split out; and thus, system **70** is an asymmetric multi-processing model. Interconnection medium **74** is implemented in a manner similar to interconnection medium **62** of system **60**.

[0199] Both system **60** and system **70** integrate network transport acceleration and server functionality within a common chassis. This provides cost reduction in terms of shared power supplies and physical structural components. A number of serving units may be placed in a rack, and may share the same management and interface components. Higher interconnection speeds occur within a single chassis as compared to connections between physically separate devices.

[0200] **FIG. 8** illustrates a system **80** in which security accelerator **81** is embedded on a network interface card (NIC) **80**. The NIC may be a stand alone card or incorporated with a server. The security accelerator may be used to replace the server network interface card (NIC) and perform all the responsibilities of the server NIC in addition to the security measures described herein. All network interface and security processing elements may be integrated as a single component, which may be a chip as well as a card. In system **80**, security accelerator **81** is connected to one or more processing units, such as servers, via the interconnection medium **82**. Interconnection medium **82** may be any of the various interconnection mediums described above and may include a bus type connection such as a PCI or PCI-X bus, a switch fabric, or other interconnects.

[0201] Alternatively, the security accelerator may be placed between a server NIC and the rest of the server. In this case the server NIC would pass packets to the security accelerator and the security accelerator would perform the security measures. This embodiment may be desirable for servers with more than one server NIC. It may be particularly desirable to utilized the security accelerator to manage the access control lists of a server. Servers often utilized access control lists to allow or disallow access to all files in the file systems and every file request may require a look up in the control list. Offloading the access control list management to the security accelerator will free up the other processing and memory resources of the server.

[0202] Another network endpoint system that the security techniques described herein may be utilized in is a network filer appliance. A network filer is a device that serves only a specific kind of request, such as a file requests. A filer appliance is essentially a server that has been narrowed in features in order to improve performance. It is key for a filer appliance to maximize performance. The security techniques described herein can further improve the performance of a filer appliance by offloading the access control list and ACL lookups from the other resources of the filer appliance. Freeing up CPU bandwidth and memory usage on the filer appliance results in improved performance.

[0203] As mentioned above, the intelligent security hardware/security accelerator described herein may be incorporated into a network switch or router. The switch may include a network processor that is programmed to switch packets within the network switch. This network processor or an additional network processor or other hardware may be programmed to operate as the security accelerator. Thus, an external network may be connected on one side of the switch or router and security may be provided to network nodes or endpoints connected to the other side of the switch or router. In one embodiment, the network switch may be connected to multiple clients and provide security for those clients. Programmability of the intelligent security hardware may allow the security protection to be tuned specifically to the types of attacks that may be directed at those clients. In addition, although network switches may support packet filtering, the filter size and complexity may be limited so as not to add latency to a switched packet. The techniques described herein would allow the network switch to support larger, more complex filters. Thus, full firewall functionality may be provided within the network switch.

[0204] It will be understood with benefit of this disclosure that although specific exemplary embodiments of hardware and software have been described herein, other combinations of hardware and/or software may be employed to achieve one or more features of the disclosed systems and methods. Furthermore, it will be understood that operating environment and application code may be modified as necessary to implement one or more aspects of the disclosed technology, and that the disclosed systems and methods may be implemented using other hardware models as well as in environments where the application and operating system code may be controlled.

What is claimed is:

1. A network processing system connected to a network that carries data in packet format, comprising:

a security accelerator having a processor programmed to receive packets from the network and to examine each packet to determine whether data in the packet represents a potential security violation;

at least one processing unit programmed to respond to requests contained within the packets; and

an interconnection medium for directly connecting the security accelerator to the processing units.

2. The system of claim 1, wherein the processor is a network processor.

3. The system of claim 1, wherein the processor is a CPU processor.

4. The system of claim 1, wherein the security accelerator further has hardware logic operable in conjunction with the processor.

5. The system of claim 1, wherein the interconnection medium is a bus.

6. The system of claim 1, wherein the interconnection medium is a switch fabric.

7. The system of claim 1, wherein the interconnection medium is shared memory.

8. The system of claim 1, wherein the network is the Internet.

9. The system of claim 1, wherein the network is a private network.

10. The system of claim 1, wherein the security accelerator determines a potential security violation by determining whether a packet is part of a denial of service attack.

11. The system of claim 1, wherein the security accelerator determines a potential security violation by determining whether a packet is part of a syn attack.

12. The system of claim 1, wherein the security accelerator is further programmed to determine whether outgoing data is authorized to be sent.

13. The system of claim 1, wherein the network processing system is an endpoint system.

14. The system of claim 13, wherein the endpoint system is a content delivery system.

15. The system of claim 1, wherein the network processing system is a client system.

16. The system of claim 1, wherein the network processing system is a single chassis system.

17. A method for processing network data at a network processing system that receives packet data via a network, comprising the steps of:

using a security accelerator having a processor to receive packets from the network and to examine each packet to determine whether data in the packet represents a potential security violation;

using at least one processing unit to respond to requests contained within the packets; and

directly connecting the security accelerator to the processing unit via an interconnection medium.

18. The method of claim 17, wherein the processor is a network processor.

19. The method of claim 17, wherein the processor is a CPU processor.

20. The method of claim 17, wherein the security accelerator further has hardware logic operable in conjunction with the processor.

21. The method of claim 17, wherein the interconnection medium is a bus.

22. The method of claim 17, wherein the interconnection medium is a switch fabric.

23. The method of claim 17, wherein the interconnection medium is shared memory.

24. The method of claim 17, wherein the network is the Internet.

25. The method of claim 17, wherein the network is a private network.

26. The method of claim 17, wherein the security accelerator determines a potential security violation by determining whether a packet is part of a denial of service attack.

27. The method of claim 17, wherein the security accelerator determines a potential security violation by determining whether a packet is part of a syn attack.

28. The method of claim 17, wherein the security accelerator is further programmed to determine whether outgoing data is authorized to be sent.

29. The method of claim 17, wherein the network processing system is an endpoint system.

30. The method of claim 17, wherein the endpoint system is a content delivery system.

31. The method of claim 17, wherein the network processing system is a client system.

32. A security accelerator device for use at a network node, comprising:

at least one processor programmed to receive packets from the network and to examine each packet to determine whether data in the packet represents a potential security violation;

an front end interface for connecting the security accelerator to a network; and

a back end interface for connecting the security accelerator to an interconnection medium.

33. The device of claim 32, wherein the processor is a network processor.

34. The device of claim 32, wherein the processor is a CPU processor.

35. The device of claim 32, wherein the security accelerator further has hardware logic operable in conjunction with the processor.

36. The device of claim 32, wherein the interconnection medium is a bus.

37. The device of claim 32, wherein the interconnection medium is a switch fabric.

38. The device of claim 32, wherein the interconnection medium is shared memory.

39. The device of claim 32, wherein the security accelerator, the front end interface, and the back end interface are fabricated as a single circuit component.

40. A network connectable computing system providing at least some security functions in addition to system functionality, the system being configured to be connected on at least one end to a network, the system comprising:

at least one network connection configured to be coupled to the network;

at least one system processor for performing system functionality;

security hardware located in a data path between the network connection and the at least one processor; and

an interconnection between the at least one processor and the security hardware,

wherein the security hardware off-loads at least some security functions from other system resources by analyzes data packets entering the network connectable computing system to perform security functions prior to forwarding the data packets to the remainder of the system.

41. The system of claim 40 wherein the security hardware comprises a network processor.

42. The system of claim 41 wherein the security functions are programmable.

43. The system of claim 41, wherein the analysis of data packets comprises analyzing data packet headers.

44. The system of claim 43, wherein the at least one system processor and the network processor communicate in a peer to peer environment across a distributed interconnect.

45. The system of claim 44, wherein the at least one system processor comprises at least one storage processor and at least one application processor.

46. The system of claim 40, wherein the network connectable computing system is a network endpoint system and the at least one system processor comprises at least one storage processor and at least one application processor.

47. The system of claim 46, wherein the interconnection is a switch fabric.

48. A method of operating a network connectable computing system, comprising:

receiving data from a network;

analyzing the data with programmable security hardware to decode incoming data packet headers;

performing at least one security function based upon the analysis of the data packet header; and

forwarding the data packet to at least one system processor through a system interconnection after performing the at least one function.

49. The method of claim 48 wherein the security function is to determine if the data is part of a security attack.

50. The method of claim 48 wherein the security function is a filtering operation.

51. The method of claim 48 wherein the security function is an authentication verification or access control list function.

52. The method of claim 48, further comprising performing security functions on outgoing data packets to provide bi-directional security functionality.

53. The method of claim 48 wherein the security function is performed by a network processor.

54. The method of claim 53 wherein the network connectable computing system is a network endpoint system.

55. The method of claim 54 wherein the network processor is programmable to allow the implementation of different security algorithms.

56. A network endpoint system for performing endpoint functionality, the endpoint system comprising:

at least one system processor, the system processor performing endpoint processing functionality;

a distributed interconnect coupled to the at least one system processor; and

security hardware coupled to the distributed interconnect,

wherein the system is configured such that a data packet from a network may be processed by the security hardware prior to being processed by the at least one system processor, and

wherein the security hardware is configured to process at least a portion of the data packet to perform a security function prior to the security hardware forwarding the data packet to the distributed interconnect.

57. The network endpoint system of claim 56, wherein the security hardware is programmable so that different security algorithms may be implemented in the security hardware.

58. The network endpoint system of claim 56, wherein the at least one system processor comprises at least one storage processor and at least one application processor.

59. The network endpoint system of claim 58, wherein the security hardware comprises at least one network processor.

60. The network endpoint system of claim 59, wherein the network processor, the storage processor and the application processor operate in a peer to peer environment across the distributed interconnect.

61. The network endpoint system of claim 60, wherein the distributed interconnect is a switch fabric.

62. The network endpoint system of claim 56, wherein the network endpoint system is a content delivery system.

63. The network endpoint system of claim 62 wherein:

the security hardware comprises at least one network processor;

the at least one system processor comprises at least one storage processor and at least one application processor, the storage processor being configured to interface with a storage system; and

the network processor, the storage processor and the application processor operate in a peer to peer environment across the distributed interconnect.

64. The network endpoint system of claim 63 wherein the distributed interconnect is a switch fabric.

65. The network endpoint system of claim 64, wherein the system is configured in a single chassis.

66. A method of operating a network endpoint system, comprising:

providing a network processor within the network endpoint system, the network processor being at an interface which couples the network endpoint system to a network;

processing data passing through the interface with the network processor;

performing security functions as part of the processing of the network processor; and

forwarding incoming network data from the network processor to a system processor which performs at least some endpoint functionality upon the data.

67. The method of claim 66 wherein the network processor rejects incoming network data which violates security algorithms and forwards incoming data which passes security algorithms to the system processor.

68. The method of claim 66 wherein the network processor analyzes headers of data packets to perform the security functions.

69. The method of claim 68 wherein the network processor is programmable to implement different security algorithms.

70. The method of claim 68 wherein the security functions include detecting a security 70.

71. The method of claim 70 wherein the security attack comprises a denial of service attack.

72. The method of claim 68 wherein the security function is a filtering operation.

73. The method of claim 68 wherein the security function is an authentication verification or access control list function.

74. The method of claim 68, wherein the security function is performed upon outgoing data.

75. The method of claim 74 wherein the security function comprises performing security functions upon both outgoing and incoming data.

76. A network connectable computing system, comprising:

a first connection to receive data packets from a network;

security hardware comprising at least one network processor, the security hardware coupled to the interface connection; and

a second connection to transmit data processed by the security hardware,

wherein the at least one network processor analyzes at least a portion of the data packets to perform at least one security function.

77. The system of claim 76, wherein the network processor analyzes headers of the data packets.

78. The system of claim 76, wherein the system is an intermediate network node system.

79. The system of claim 78, wherein the system is a network switch.

80. The system of claim 76, wherein the system is a network endpoint system.

81. The system of claim 76, wherein the system is a network endpoint system having at least one server or at least one server card coupled to the second connection.

82. The system of claim 76, wherein the system is incorporated into a network interface card.

83. The system of claim 81, wherein the second connection is a distributed interconnection.

84. The system of claim 83, wherein the distributed interconnection is a switch fabric.

85. The system of claim 76, wherein the second connection is coupled to an asymmetric multi-processing system.

86. The system of claim 85, wherein the second connection is a distributed interconnection and the asymmetric multi-processing system includes a plurality of task specific processors.

87. The system of claim 86, wherein the distributed interconnection is a switch fabric and the task specific processors include storage or application processors.

88. The system of claim 87, wherein the task specific processors include storage and application processors.

* * * * *

US006412009B1

(12) **United States Patent**
Erickson et al.

(10) **Patent No.: US 6,412,009 B1**
(45) **Date of Patent: Jun. 25, 2002**

(54) **METHOD AND SYSTEM FOR PROVIDING A PERSISTENT HTTP TUNNEL**

(75) Inventors: **Rodger D. Erickson**, St. Louis, MO (US); **Ronald D. Sanders**, Spokane, WA (US)

(73) Assignee: **Wall Data Incorporated**, Kirkland, WA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **09/268,068**

(22) Filed: **Mar. 15, 1999** 102(e).

(51) **Int. Cl.**[7] **G06F 13/00**

(52) **U.S. Cl.** **709/228**; 709/217; 709/236; 709/313

(58) **Field of Search** 709/203, 217, 709/219, 223, 224, 225, 227, 228, 236, 238, 313

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,754,830 A	*	5/1998	Butts et al.	395/500.44
5,778,372 A	*	7/1998	Cordell et al.	707/100
5,935,212 A	*	8/1999	Kalajan et al.	709/228
5,941,988 A	*	8/1999	Bhagwat et al.	713/201

OTHER PUBLICATIONS

Provan, D., "Tunneling IPX Traffic through IP Networks," RFC 1234, Novell, Inc. (Jun. 1991).
Woodburn, R. et al., "A Scheme for an Internet Encapsulation Protocol: Version 1," RFC 1241, University of Delaware (Jul. 1991).

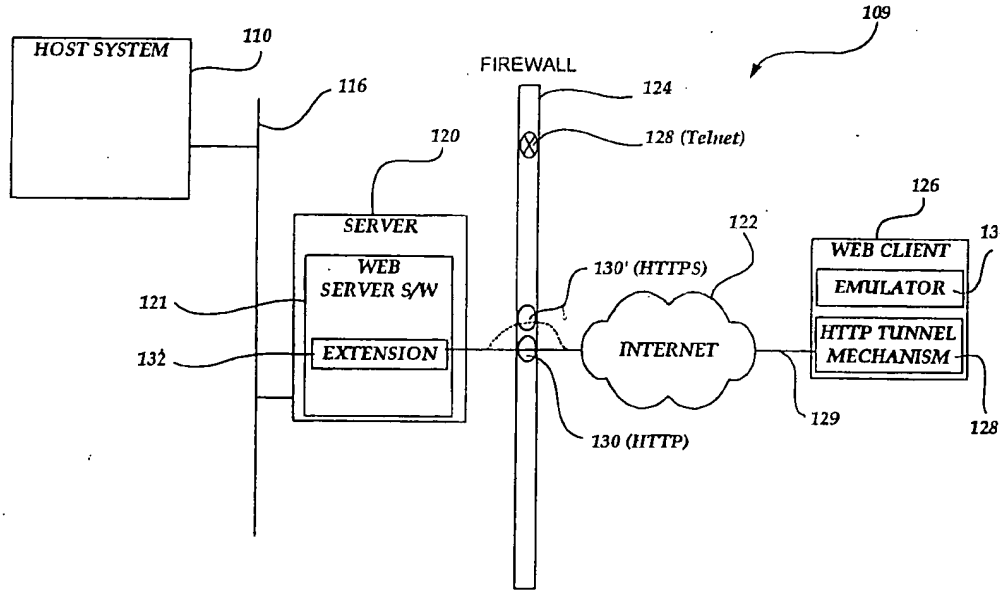
* cited by examiner

Primary Examiner—Viet D. Vu
(74) *Attorney, Agent, or Firm*—Christensen O'Oconnor Johnson Kindness PLLC

(57) **ABSTRACT**

A method and system for providing a persistent HTTP tunnel for a connection-oriented protocol between a client and a Web server. A data message complying with the connection-oriented protocol is generated and embedded into a chunked data message in compliance with a chunking option for the HTTP. The chunked data message is transmitted between a client and a Web server. Upon receiving any chunked data message at the Web server, the Web server parses the chunked data message and delivers the data message to a host system. Upon receiving any chunked data message at the client, the client parses the chunked data message and delivers the data message to a terminal emulator running on the client. This allows a terminal session to be supported by a real-time bi-directional persistent connection with the host system. The bi-directional persistent connection allows interleaving of the chunked data messages from the Web client with the chunked data messages on the Web server on the persistent HTTP tunnel.

8 Claims, 5 Drawing Sheets



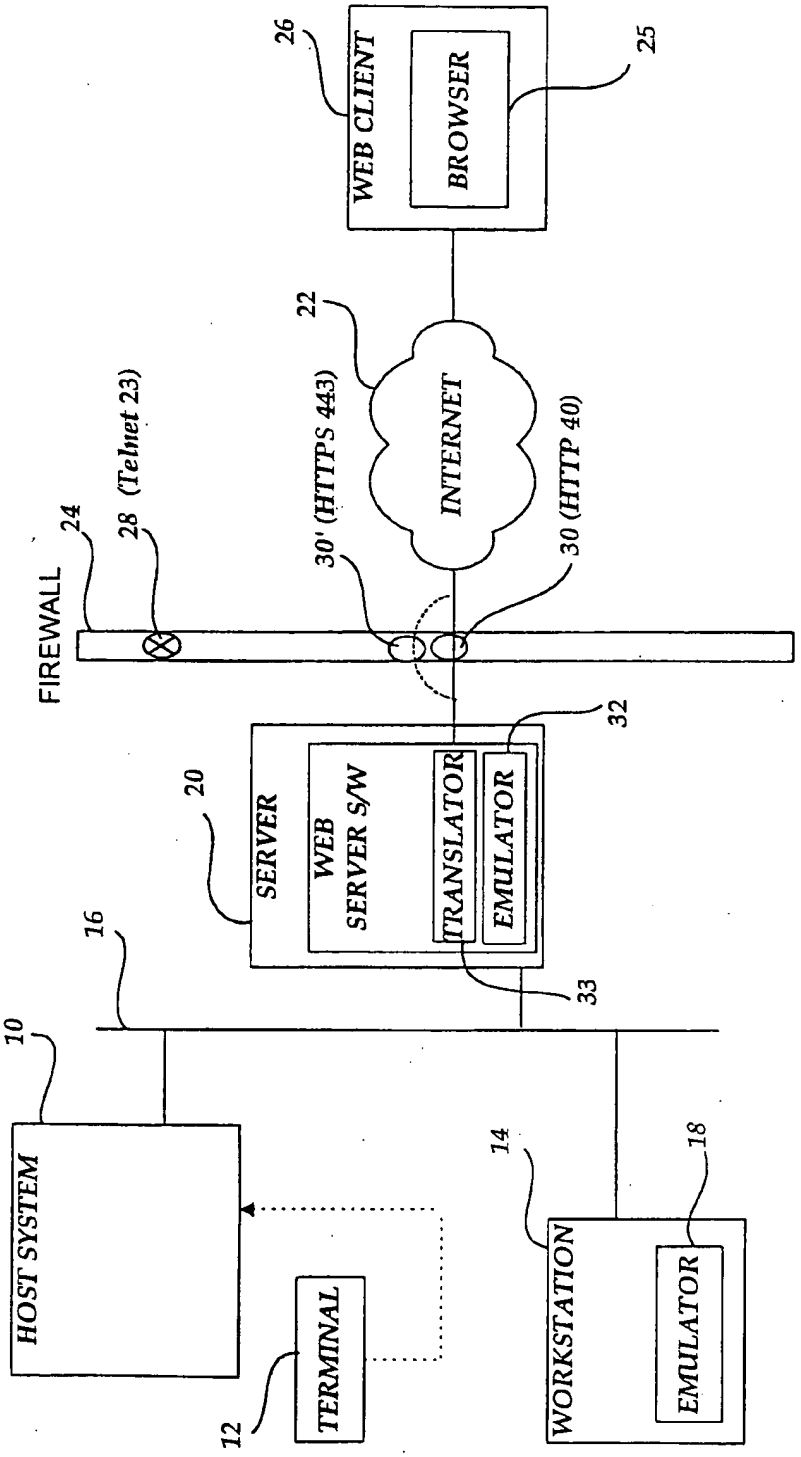


Fig.1.
PRIOR ART

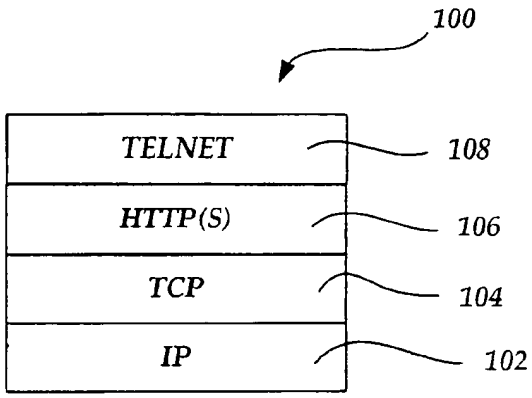


Fig.2.

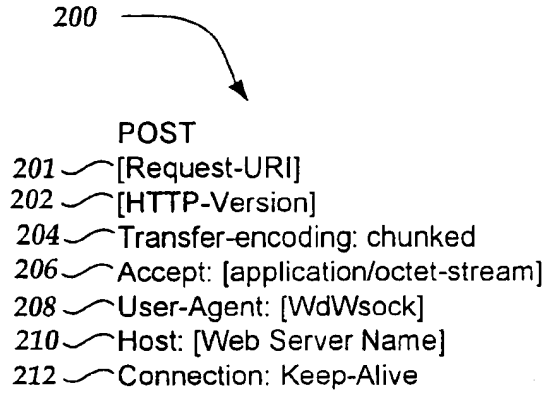


Fig.5.

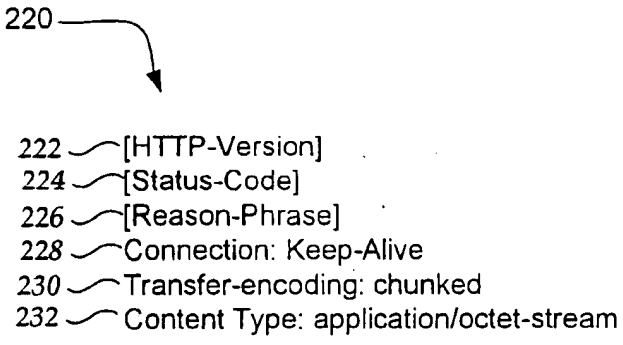


Fig.6.

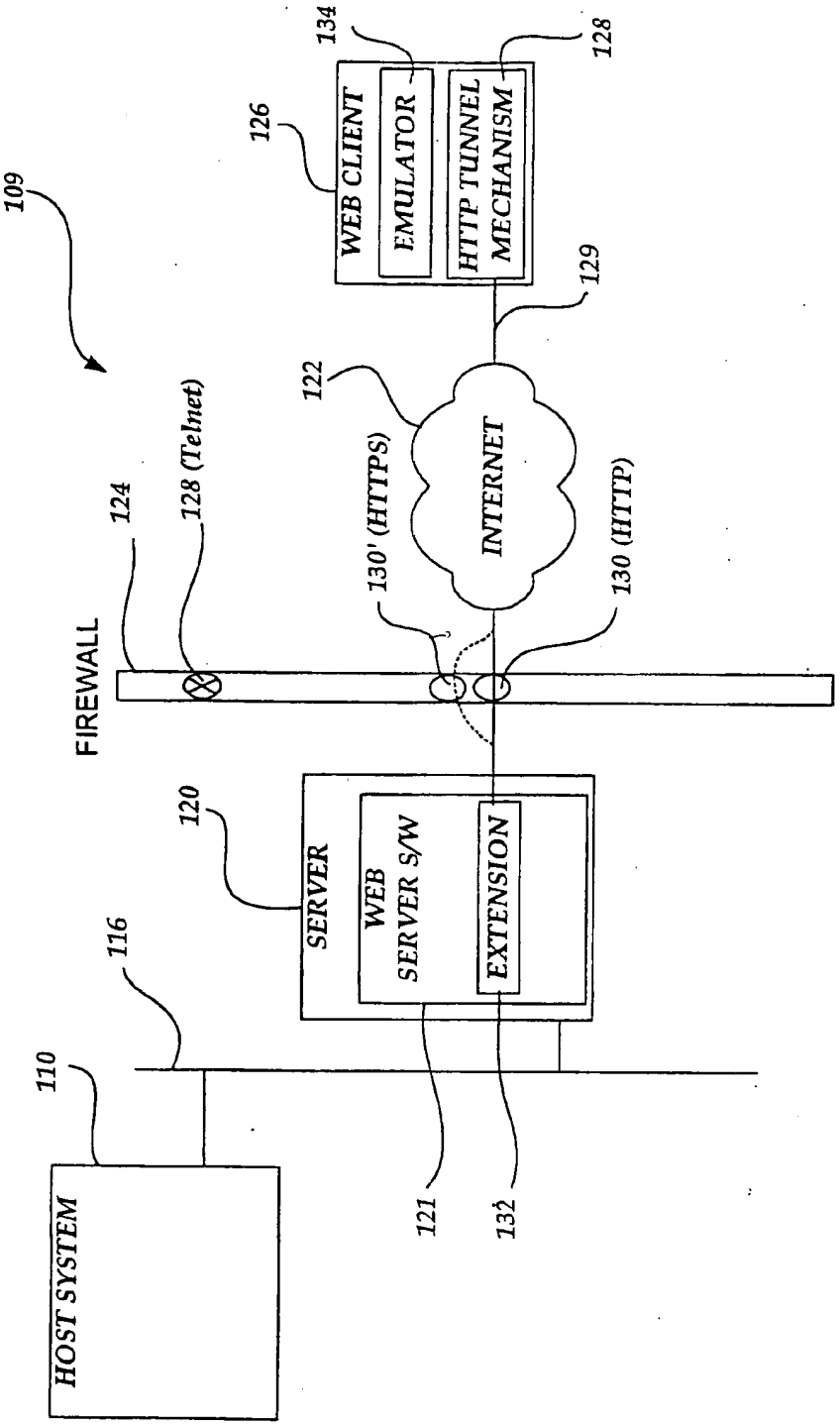

109
WEB CLIENT
EMULATOR
HTTP TUNNEL MECHANISM
126
134
128
129
122
INTERNET
128 (Telnet)
124
130' (HTTPS)
130 (HTTP)
FIREWALL
120
SERVER
WEB SERVER S/W
EXTENSION
116
110
HOST SYSTEM
121
132

Fig.3.

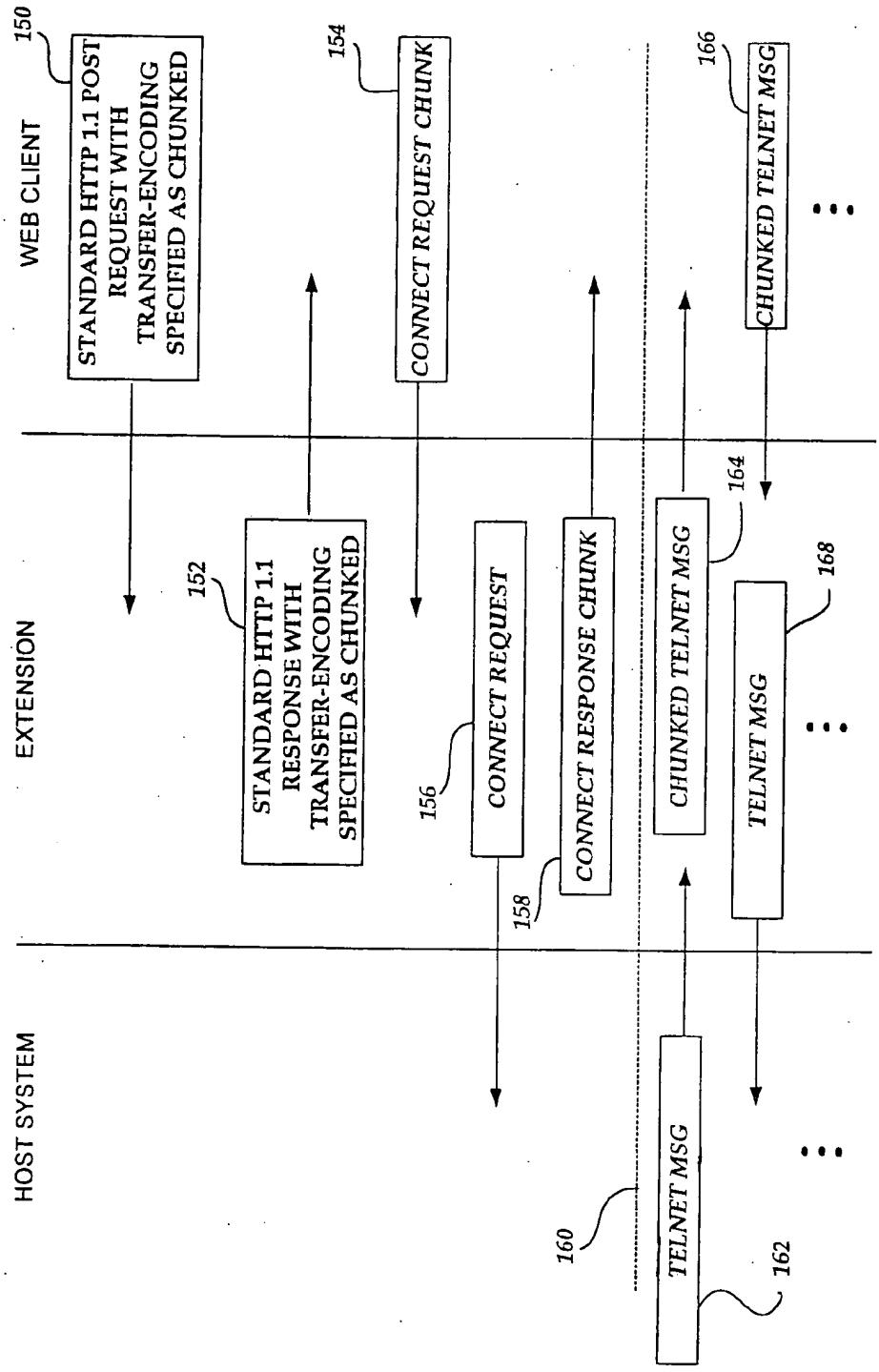

WEB CLIENT
EXTENSION
HOST SYSTEM
150
STANDARD HTTP 1.1 POST REQUEST WITH TRANSFER-ENCODING SPECIFIED AS CHUNKED
152
STANDARD HTTP 1.1 RESPONSE WITH TRANSFER-ENCODING SPECIFIED AS CHUNKED
154
CONNECT REQUEST CHUNK
156
CONNECT REQUEST
158
CONNECT RESPONSE CHUNK
160
TELNET MSG
162
CHUNKED TELNET MSG
164
TELNET MSG
166
CHUNKED TELNET MSG
168

Fig.4.

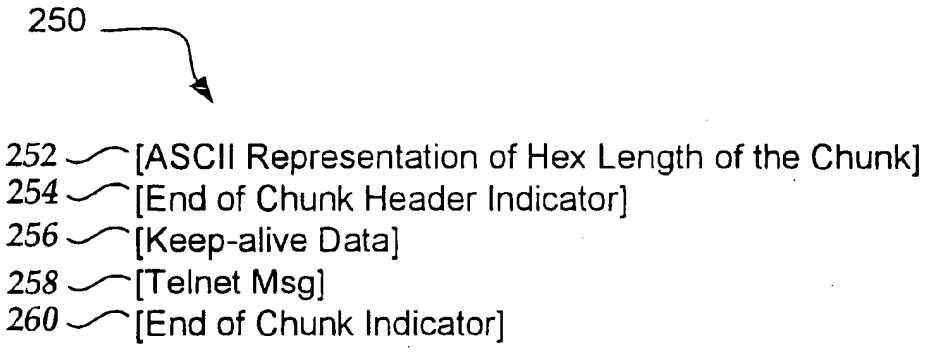


Fig.7.

PRIOR ART

METHOD AND SYSTEM FOR PROVIDING A PERSISTENT HTTP TUNNEL

FIELD OF THE INVENTION

The present invention relates generally to communications between two computers, and, more particularly, to a communication method and system that provides a persistent HTTP tunnel for a connection-oriented protocol between the two computers.

BACKGROUND OF THE INVENTION

Many corporations continue to maintain their corporate computer data on what are referred to as "legacy host" systems. These systems are generally older mainframe or mini-computers which cannot be easily replaced. As personal computers become more commonplace, a significant effort has been devoted to methods by which a user of a personal computer can access data stored on a legacy host. One such system is shown in FIG. **1**, whereby access to a host system **10** may be through a terminal **12** directly coupled to the host system **10** that provides a local login capability. If a workstation **14** desires to connect to the host system **10** over a Local Area Network (LAN) **16**, the workstation **14** runs a terminal emulator **18**. A Telnet protocol is used between the host system **10** and the workstation **14** to provide the workstation **14** access to the host system **10** as if the workstation was a local terminal directly connected to the host system.

With the phenomenal growth of the Internet, a need developed to provide the ability of a user to access a host computer from anywhere in the world. A typical configuration to allow such Internet access includes a Web server **20** coupled to the host system **10** via the LAN **16**. The Web server **20** is then coupled to the Internet **22** through a firewall **24**. The firewall **24** enforces a security policy between a secure internal network containing the host system **10** and an untrusted network such as the Internet. The firewall may be a personal computer (PC), a router, a mainframe, a UNIX workstation, or any combination of such devices responsible for enforcing the security policy. A Web client computer **26** runs a browser program **25** to access the Web server **20** through the Internet **22**.

Because the Internet **22** is an unsecured network, most firewall security policies will not allow the Web client **26** to communicate using the well-known "Telnet port 23," shown at **28**. However, most firewalls allow communications through the well-known "HyperText Transfer Protocol (HTTP) port 80," shown at **30**, and the secure "HTTP port 443," shown at **30'**. Therefore, a current system that provides a local login experience to a Web client **26** uses one of these HTTP ports **30** **30'**. In this system, the Web server **20** runs a terminal emulator **32** that provides a Telnet session with the host system **10**. The Web server **20** receives the Telnet data from the host system **10** and instead of displaying the data as a typical text screen will instead send the Telnet data to a translator **33**. The translator **33** translates the Telnet data to HyperText Markup Language (HTML) statements that are sent to the browser program **25** running on the Web client **26**. The browser **25** then translates the HTML statements and displays an HTML page on the Web client.

A problem with this type of system is that the translated HTML screen does not look sufficiently similar to a local login screen that the user would see if they were directly connected to the host system and the interaction with the HTML screen is not sufficiently similar to the interaction with the directly connected terminal. In certain situations, the differences may require additional training of the users on the Web client. Another problem is that the response times between a user request on the Web client **26** and the return response from the host system is more variable in comparison with a local login response time. For example, response times may range from a second to thirty seconds using the HTML screen, in comparison with response times in the range of one second to three seconds with a directly connected terminal. The variable response times are due to the nature of the HTTP protocol.

HTTP is a request/response protocol. The Web client **26**, using the browser **25**, establishes a connection with the Web server **20** and sends a request to the Web server. After the Web server sends a response to the browser **25**, the connection is closed. Before additional requests may be handled, a new connection must be established. Even though the newer HTTP 1.1 specification provides a keep-alive mechanism that allows one connection for multiple objects on an HTML page, the connection is closed either by the Web server or the browser after a period of inactivity. The period of inactivity may range from several seconds to a fraction of a second depending on the activity on the Web server. Many protocols, including Telnet, have insufficient transaction rates to maintain an alive connection even when the Web server is only modestly loaded. Closing the connection and establishing a new connection creates significant overhead resulting in decreased performance.

Given the shortcomings associated with the prior art method of providing access to host computer systems for Web clients, there is a need for a method that uses existing standard ports in the firewall while providing more consistent response times similar to the response times of a workstation connected through a LAN to a host system. The present invention is directed to filling this need.

SUMMARY OF THE INVENTION

In accordance with this invention, a server, a client, and a method of operation are provided for a Web client to access a host system with performance and displays comparable to the performance and displays of a workstation connected through a LAN to the host system.

In accordance with one aspect of this invention, a method of providing a persistent HTTP tunnel for a persistent virtual session is provided. A data message complying with a connection-oriented protocol is generated at an endpoint of a connection-oriented virtual session. The data message is embedded into a chunked data message complying with a chunking option of an HTTP specification. The chunked data message is transmitted between a Web client and a Web server via an HTTP connection. Upon receiving any chunked data message at the Web server, the Web server parses the chunked data message and delivers the data message to one endpoint of the connection-oriented virtual session. In the other direction, upon receiving any chunked data message at the Web client, the Web client parses the chunked data message and delivers the data message to another endpoint of the connection-oriented virtual session. The chunked data messages from the Web client are interleaved with the chunked data messages from the Web server on the persistent HTTP tunnel.

In accordance with other aspects of this invention, the connection-oriented protocol is a Telnet protocol.

In accordance with still further aspects of this invention, one endpoint of the connection-oriented virtual session is a host system.

In accordance with yet further aspects of this invention, the other endpoint of the connection-oriented session is a Web client application.

In accordance with still other aspects of this invention, the Web client application is a terminal emulator.

In accordance with another aspect of this invention, a method for creating a persistent tunnel between a Web client and a Web server using an HTTP protocol for providing a persistent virtual connection between a host system and the Web client is provided. A connection between the Web client and the Web server is established using a chunking option in accordance with an HTTP protocol that allows a series of messages to be sent as chunked messages. A virtual session is established between the host system and the Web client through a Web Server extension. A plurality of host messages are transmitted from the host system to the Web server extension and inserted into a chunked host message at the Web server. The Web server forwards the chunked host messages to the Web client over the connection. The Web client parses the chunked host message and delivers the host message to an application. In the other direction, a plurality of Web client messages are transmitted from the application to a tunneling mechanism on the Web client. The tunneling mechanism inserts the client message into a chunked client message and forwards the chunked client message to the Web server over the connection. The Web server forwards the chunked client message to an extension before receiving subsequent chunked client messages. The extension parses the chunked client message and delivers the client message to the host system. The chunked data messages from the Web client are interleaved with the chunked data messages from the Web server on the persistent HTTP tunnel.

In accordance with a further aspect of the present invention, a server for providing a persistent virtual session over HTTP is provided. The server includes a server software component operable to communicate via a persistent HTTP tunnel with a first endpoint of a connection-oriented session. The server also includes an extension operable to communicate with the server software component and a second endpoint of a connection-oriented session. Upon a connect request from a client, the extension establishes a connection-oriented session with the second endpoint to provide a virtual connection-oriented session between the first endpoint and the second endpoint. After the virtual connection-oriented session is established, the extension receives one or more chunked client messages from the client. The chunked messages comply with a chunking option as specified in the Hyper Text Transfer Protocol. Each chunked client message includes a chunk header and a data portion. The extension forwards the data portion to the second endpoint over the connection-oriented session. The extension also receives one or more second endpoint messages from the second endpoint and encapsulates each second endpoint message into a chunked second endpoint message. The extension then forwards the chunked second endpoint message to the client that delivers the second endpoint message of the chunked second endpoint message to the first endpoint.

In accordance with other aspects of this invention, the second endpoint is a host system.

In accordance with still further aspects of this invention, the first endpoint is a client application.

In accordance with yet still further aspects of this invention, the client application is a terminal emulator.

In accordance with a further aspect of this invention, a client having a first endpoint of a connection-oriented session having a persistent virtual session with a second endpoint over HTTP is provided. The client includes an application for sending and receiving data messages complying with the connection-oriented session at the second endpoint and an HTTP tunnel mechanism. The HTTP tunnel mechanism receives the data messages generated by the application and inserts the data messages into a chunked data message complying with HTTP and transmits the chunked data messages to a Web server. The HTTP tunnel mechanism also receives chunked data messages generated by the Web server and forwards the data messages within the chunked data message to the application.

A technical advantage of the present invention is the ability to establish a connection-oriented virtual session between a host system and a Web client through the commonly available HTTP port. Both the Web server and the Web client encapsulate the connection-oriented session data, such as Telnet, into chunks that comply with the HTTP specification. The Web server transmits the HTTP response as soon as the request is received and transmits chunked messages and receives chunked messages from the Web client in an interleaving manner without completing the original request. Because the chunks are interleaved, the present invention provides a persistent bi-directional virtual connection between the host system and the Web client.

BRIEF DESCRIPTION OF THE DRAWINGS

The foregoing aspects and many of the attendant advantages of this invention will become more readily appreciated as the same become better understood by reference to the following detailed description, when taken in conjunction with the accompanying drawings, wherein:

FIG. 1 is a block diagram of a prior art method for connecting Web clients to host computer systems;

FIG. 2 is a block diagram of a protocol stack for tunneling Telnet data on HTTP;

FIG. 3 is a block diagram of a system for connecting Web clients to host systems in accordance with the present invention;

FIG. 4 is a block diagram illustrating a communication flow among a Web client, an extension residing on a Web server, and a host system in accordance with the present invention;

FIG. 5 is a format for an HTTP Post request suitable for use in the present invention;

FIG. 6 is a format for an HTTP response message to the HTTP Post request of FIG. 5 suitable for use in the present invention; and

FIG. 7 is a format of an HTTP chunk message suitable for use in the present invention.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

In accordance with the present invention, access to host systems from Web clients is available in a manner that provides performance and displays comparable to the performance and displays of a workstation connected through a LAN to the host system. The present invention provides a system and method for providing a persistent HTTP tunnel for any connection-oriented protocol and keeping the HTTP tunnel connection between the Web client and the Web server persistent for the duration of the communication between the Web client and host system. By providing a persistent connection for the entire duration, the present invention achieves performance comparable to the performance of the workstation 14 connected through the LAN 16 to the host system, as shown in FIG. 1. In addition, the present invention utilizes an existing port in the firewall rather than requiring a new hole in the firewall.

Although the presently preferred embodiment of the invention utilizes the connection-oriented protocol Telnet, those skilled in the art will recognize that other connection-oriented protocols may be used. As mentioned earlier, the Telnet protocol is typically used between a host system and a client residing on a network to provide the client access to the host system as if the client was a local terminal directly connected to the host system.

FIG. **2** is a block diagram illustrating how Telnet data is tunneled through HTTP in accordance with the present invention. A protocol stack **100** of the present invention has a first or lower layer **102** that represents an Internet layer or a network layer that shields higher layers **104–108** from a physical network architecture. An Internet Protocol (IP) is the protocol of the first layer **102** and is a connectionless protocol that does not provide reliability, flow control or error recovery. The IP provides a routing function that ensures that messages will be correctly delivered to their destination. A second layer **104** is a transport layer that provides an end-to-end reliable data transfer. The second layer **104** allows multiple applications to be supported simultaneously. A Transmission Control Protocol (TCP) is used in the second layer **104** to provide a reliable exchange of information. A third layer **106** is an application layer. An interface between the application layer and the transport layer is defined by port numbers and sockets. In contrast to prior art methods of implementing Telnet on top of TCP, the present invention tunnels Telnet on top of the HTTP(S) protocol of the third layer **106** resulting in a fourth layer **108** of the protocol stack **100**. One skilled in the art will appreciate that HTTP is considered the "official" application protocol of the World Wide Web and Telnet is considered the "official" protocol for emulating a remote terminal. The tunneling of the Telnet protocol on top of the HTTP protocol will be described in greater detail below.

FIG. **3** is a block diagram of a system for connecting Web clients to host systems in accordance with the present invention. A system **109** includes a Web server **120** coupled to a host system **110** via a LAN **116**. The Web server **120** is also coupled to an Internet **122** through a firewall **124**. The firewall **124** enforces a security policy between a secure internal network containing the host system **110** and an untrusted network such as the Internet. The firewall **124** may be a personal computer (PC), a router, a mainframe, a UNIX workstation, or any combination of such devices responsible for enforcing the security policy.

A Web client **126** is coupled to the Web server **120** through the Internet **122**. The system **109** further comprises an extension **132** residing on the Web server **120**. The extension **132** may run in the same process space as Web server software **121** or communicate with the Web server software **121** using an interprocess communication well-known in the art. The Web server software **121** passes all commands and data designating the extension **132** to the extension **132**. The commands and data are received from the Web client through a HyperText Transfer Protocol (HTTP) port **80**, shown at **130**, or the secure HTTP port **443**, shown at **130'** in the firewall **124**. As one skilled in the art will appreciate, the firewall **124** passes the commands and data from the Web client onto the Web server **120**. Because the present invention utilizes the standard HTTP ports **130** and **130'**, additional holes in the firewall are not necessary. In addition, because the present invention utilizes the standard HTTP ports, the present invention minimizes the amount of development necessary to create a secure Web server environment. Rather, the present invention can utilize existing resources residing on the Web server, thereby leveraging their development costs. The extension **132** is responsible for providing a virtual session between the host system **110** and the Web client **126**.

The Web client **126** comprises an HTTP tunnel mechanism **128** and an emulator **134**. The HTTP tunnel mechanism **128** is a software module that provides an Application Programming Interface (API) for TCP, such as Winsock created by Microsoft Corporation, of Redmond, Wash., or WDWSock created by Wall Data Incorporated, of Kirkland, Wash. The HTTP tunnel mechanism **128** receives messages created in the extension **132**, parses the messages, and passes any Telnet data or commands to the emulator **134** so that the emulator **134** can emulate a host terminal.

The present invention utilizes a chunking option as specified in the HyperText Transfer Protocol specification 1.1 dated January 1997, referred to as HTTP/1.1 (RFC 2068), the specification of which is hereby incorporated by reference. The chunking option refers to a new transfer coding implementation that allows a body of a message to be transferred as a series of chunks, "chunked data," each with its own size indicator, followed by an optional footer containing entity-header fields. Prior to the chunking option, the size of the message along with the entire message must be sent at the same time. The present invention expands the use of the chunking option by sending a series of HTTP messages with embedded session-oriented data, "chunked messages," between the Web server and the Web client without sending an end chunk message. This allows one connection to remain active during the series of interleaved HTTP messages between the Web server and the Web client, thereby creating an HTTP tunnel **129** that is persistent for the duration of the communication between the host system and the Web client. The interleaved HTTP messages include messages that alternate between the Web client and the Web server as the sender for each message or messages that alternate between one or more Web client messages and then one or more Web server messages. Therefore, a persistent virtual connection between two endpoints of the connection-oriented protocol is provided. One endpoint located on the host system and the other endpoint on the Web client. In the embodiment described, a terminal emulator is the endpoint on the Web client and a Telnet process is the endpoint on the host system.

Depending upon the type of Web server **120**, a setting may need to be changed to inform the Web server **120** not to process chunked data designating the extension **132**. By changing the setting, the extension is allowed to handle the chunked data in a manner described in detail below. To change the setting of any Web server **120** running Microsoft Internet Information Server (IIS) 4.0 by Microsoft Corporation, the extension **132** accesses a metabase and programmatically changes an MD_UPLOAD_READAHEAD_SIZE parameter to zero (0) for the named extension **132**. The MD_UPLOAD_READAHEAD_SIZE parameter is typically used to fine tune memory usage on the Web server **120**. Typically, the MD_UPLOAD_READAHEAD_SIZE parameter is set to a value that is an average size of a request message received from a Web client **126**. The present invention therefore modifies the MD_UPLOAD_READAHEAD_SIZE parameter for an unusual purpose of having the Web server software **121** ignore processing of chunked data that is designated to be passed to the extension **132**. By disabling the processing of chunked data in the Web server software **121**, the extension **132** can provide the processing for the chunked data, thereby allowing a connection-oriented protocol to be tunneled on top of HTTP and creating a bi-directional connection that

remains active for the duration of the communication between a Web client and a host system.

FIG. **4** is a block diagram illustrating the communication flow among the Web client **126**, the extension **132**, and the host system **110**. Again, Telnet is used as the connection-oriented protocol and is given by way of nonlimiting example only. At block **150**, the Web client **126** generates a standard HTTP/1.1 post request with transfer-encoding specified as chunked. The creation of the post request will be described in greater detail below. At block **152**, the extension receives the post request, and generates a standard HTTP/1.1 response with transfer-encoding specified as chunked which is sent back to the Web client. At block **154**, the Web client generates and sends a connect request chunk to the extension. The connection request chunk includes a destination port field for specifying the port number of the host system for connection. In the above-described embodiment, the destination port field contains a value specifying the Telnet port **23**. The connect request chunk also includes a destination IP address field for specifying the HP address for the host system. Because the connect request chunk occurs after the initial post response, the connect request chunk uses the format for chunked data. At block **156**, the extension sends a connect request to the host system to establish a Telnet session. At block **158**, the extension generates and sends a connect response chunk to the Web client. After the Web client receives the connect response chunk, the end-to-end session between the Web client and the host system via the extension is active.

The communication flow occurring after the reference line **160** represents Telnet data tunneled over HTTP. A typical exchange has the host system starting Telnet negotiation. At block **162**, the host system generates a Telnet message. The Telnet message includes Telnet commands and/or Telnet data and is sent to the extension. At block **164**, the extension encapsulates the Telnet message into a chunked Telnet message by embedding the Telnet message in a chunk that complies with the HTTP/1.1 format with Transfer-Encoding specified as chunked. The chunked Telnet message is sent via the HTTP tunnel **129** through the HTTP port over the Internet to the Web client.

The HTTP tunnel mechanism **128** in the Web client **126** parses the chunked Telnet message and provides the emulator **134** the original Telnet message as sent from the host system. Whenever the emulator generates a Telnet message, the HTTP tunnel mechanism **128** creates a chunked Telnet message at block **166** and forwards the chunked Telnet message through the Internet to the Web server software that in turn passes the chunked Telnet message to the extension **132**. The extension unchunks the chunked Telnet message and forwards to the host system only the Telnet message as originally sent by the emulator to the HTTP tunnel mechanism, at block **168**. As indicated, the sending of Telnet messages and the chunking of the Telnet message into a chunked Telnet message continues without having a new connection established between the Web client and the Web server. Because only one connection is needed during the communication flow, the present invention provides performance comparable to workstations connected through a LAN to a host system (shown in FIG. **1**) and responses that are less variable than response times of prior art methods using translated HTML statements.

One aspect of the present invention is enabling the Web server software **121** to pass the chunked messages through to the extension without waiting for additional chunked messages prior to delivery to the extension **132**. This allows the present invention to tunnel Telnet messages on top of the HTTP protocol using the standard HTTP ports **130**, **130'** and to use the HTTP ports **130**, **130'** as a bi-directional connection for a virtual session between the host system and the Web client.

FIGS. **5–7** illustrate formats for the communication flows of FIG. **4** in more detail. First, FIG. **5** illustrates a standard HTTP/1.1 post request format suitable for use in the present invention. The format of the post request is as specified in the HyperText Transfer Protocol HTTP/1.1 specification. The present invention is concerned with the contents of the post request rather than the format of the post request. At line **201**, a request-URI field contains a name for the extension **132** in the Web server. The name in the request-URI field may include a dynamic link library (DLL) or other name of a component depending on the operating system platform. At line **202**, an HTTP-version field must specify version 1.1 or greater. Only version 1.1 or greater allows a transfer-encoding to be chunked as is necessary for the present embodiment of the invention. On line **204**, the transfer-encoding is designated as chunked.

As illustrated in FIG. **4**, at blocks **154, 158, 164,** chunking allows the Web client and the host system to exchange a series of messages without having to open a new connection. At line **206** of FIG. **5**, in the present embodiment, the Web server is requested to accept application/octet-stream data. This is not necessary for the purposes of the present invention but is typical for Web servers to expect binary data. At line **208**, in the User Agent field, an identifier for the HTTP tunnel mechanism is provided. The identifier is shown as a non-limiting example and is not necessary to practice the present invention. At line **210**, in the Host field, the name of the Web server is specified. At line **212**, in the Connection field, Keep-Alive is specified so that the connection remains active. However, even with Keep-Alive specified, the connection may be terminated by either the Web client or the Web server after even very brief periods of idleness. In order to achieve a true Keep-Alive connection, the present invention incorporates additional Keep-Alive data in the tunneled data as will be described in greater detail with reference to FIG. **7**.

FIG. **6** illustrates a format for a standard HTTP/1.1 response. Again, the format of the data is as specified in the HTTP/1.1 specification and the present invention is only concerned with the content of the fields. Lines **222**, **224**, and **226** are generated by the Web server in response to the post request (FIG. **5**) that was received and are explained in the HTTP/1.1 specification. Lines **228**, **230**, and **232** are generated by the extension before the response is sent to the Web client. At line **228**, the connection is specified as Keep-Alive as described earlier. Transfer encoding is specified as chunked and the Web server indicates the format of response data as application/octet-stream (i.e., binary data rather than text or other special format data).

FIG. **7** illustrates a format for a chunk suitable for use in the present invention. At line **252**, the first field specifies an ASCII representation of the hex length of the chunk. According to the HTTP 1.1 specification, by allowing each chunk to specify the hex length of the chunk, multiple chunks (messages) may be sent for one HTTP request. However, the last message must provide a unique identifier to indicate when the HTTP request is complete. Typically, the Web server software waits until all the chunks for one message body are received before generating a response. However, because the present invention uses the chunking option in a new manner and does not send a last message, the present invention prevents the Web server software from buffering the chunked messages and instead forces the Web

server software to send the chunked messages to the extension as the chunked messages are received. This allows the present invention to provide bi-directional communication without waiting for an entire message and without establishing more than one connection. At line **254**, an end of chunk header indicator is chosen according to the HTTP/1.1 specification. Lines **256** and **258** provide the tunneling aspect of the present invention. First, line **256** provides a data field that may be sent in all chunked packets to keep the connection alive. This Keep-Alive field is necessary because some Web servers will close a connection if additional chunks are not received within a certain time frame. Therefore, in the present invention, the HTTP tunnel mechanism that is running on the Web client generates a chunk with only the Keep-Alive data field and without any additional Telnet data after a predetermined period of inactivity. The sending of only the Keep-Alive data in a chunk allows the connection to remain alive even during periods of inactivity. Line **258** represents the tunneling of the Telnet messages generated by either the host system or the Web client emulator. The Telnet messages generated by either the host system or the Web client emulator are written into the Telnet message field. Line **260** specifies the end of chunk indicator as specified in the HTTP/1.1 specification.

The connection between the Web client and the host system is terminated when the Web client closes the HTTP tunnel mechanism **128**. The extension will also recognize when the host system has prematurely ended the session and will allow the Web server to close the HTTP tunnel.

Although the present embodiment utilizes an emulator that runs as a stand-alone program on the Web client, it will be appreciated that the functionality of the emulator and/or the HTTP tunnel mechanism may be incorporated into a browser. In another embodiment, the Web client may initiate a HEAD response/request before establishing a connection. The initiation of a HEAD response/request is necessary when a Web server needs authentication of the Web client access. The HEAD response provides the Web client with information about authentication that the Web client may incorporate into the post request and the subsequent chunked data.

As one skilled in the art will appreciate, Telnet has many variations such as TN5250, NVT, VT220, and TN3270. These and other proprietary protocols may be used without departing from the scope of this invention. Likewise, the host session may provide a terminal session such as a 5250 type terminal session, a NVT type terminal session, a VT220 type terminal session, and a 3270 type terminal session without departing from the scope of this invention. In addition, although the present embodiment included an IBM host system, it will be appreciated that other host systems, such as Hewlett-Packard and UNIX host systems may be used. Further, even though the embodiment shown provides a virtual session through the Internet, the Internet may be replaced with an intranet, WAN, or other network using HTTP.

While the preferred embodiment of the invention has been illustrated and described, it will be appreciated that various changes can be made therein without departing from the spirit and scope of the invention.

The embodiments of the invention in which an exclusive property or privilege is claimed are defined as follows:

1. A method of providing a persistent HTTP tunnel for a persistent virtual session, the method comprising:

(a) generating a data message complying with a connection-oriented protocol;

(b) embedding the data message within a chunked data message complying with a chunking option of a HTTP specification;

(c) transmitting the chunked data message between a Web client and a Web server via an HTTP connection;

(d) upon receiving any chunked data message at the Web server, parsing the chunked data message and delivering the data message to one endpoint of a connection-oriented session, and upon receiving any chunked data message at the Web client, parsing the chunked data message and delivering the data message to another endpoint of the connection-oriented session; and

(e) interleaving the chunked data messages from the Web client with the chunked data messages from the Web server on the persistent HTTP tunnel.

2. The method of claim **1**, wherein the connection-oriented protocol is a Telnet protocol.

3. The method of claim **1**, wherein the one endpoint of the connection-oriented session is a host system.

4. The method of claim **3**, wherein the other endpoint of the connection-oriented session is a Web client application.

5. The method of claim **4**, wherein the Web client application is a terminal emulator.

6. The method of claim **1**, further comprising (f) transmitting a keep alive chunk message by the Web client to the Web server after a pre-determined time period in which chunked data messages were not transmitted.

7. A method for creating a persistent tunnel between a Web client and a Web server using an HTTP protocol for providing a persistent virtual connection between a host system and the Web client, the method comprising:

establishing a connection between the Web client and the Web server using a chunking option in accordance with an HTTP protocol that allows a series of messages to be sent as chunked messages;

establishing a virtual session between a host system and the Web client through an extension;

transmitting a plurality of host messages from the host system to the Web server, inserting the host message within a chunked host message at the Web server, forwarding the chunked host message to the Web client over the connection, parsing the chunked host message at the Web client and delivering the host message to an application;

transmitting a plurality of Web client messages from the application to a tunneling mechanism on the Web client, inserting the client message within a chunked client message at the Web client, forwarding the chunked client message to the Web server over the connection, parsing the chunked client message at the Web server and delivering the client message to the host system; and

interleaving the chunked data messages from the Web client with the chunked data messages from the Web server on the persistent HTTP tunnel;

wherein the transmitted chunked client message is forwarded to the extension before the Web server is forwarded the entire plurality of chunked client messages.

8. The method of claim **7**, wherein the application is a terminal emulator.

* * * * *

US 20030139975A1

(19) United States

(12) Patent Application Publication (10) Pub. No.: US 2003/0139975 A1

Perkowski (43) Pub. Date: Jul. 24, 2003

(54) METHOD OF AND SYSTEM FOR MANAGING AND SERVING CONSUMER-PRODUCT RELATED INFORMATION ON THE WORLD WIDE WEB (WWW) USING UNIVERSAL PRODUCT NUMBERS (UPNS) AND ELECTRONIC DATA INTERCHANGE (EDI) PROCESSES

(76) Inventor: Thomas J. Perkowski, Darien, CT (US)

Correspondence Address:
Thomas J. Perkowski, Esq., P.C.
Soundview Plaza
1266 East Main Street
Stamford, CT 06902 (US)

(21) Appl. No.: 10/319,387

(22) Filed: Dec. 12, 2002

Related U.S. Application Data

(63) Continuation of application No. 09/465,859, filed on Dec. 17, 1999, now abandoned, which is a continuation-in-part of application No. 09/447,121, filed on Nov. 22, 1999, and which is a continuation-in-part of application No. 09/441,973, filed on Nov. 17, 1999, which is a continuation-in-part of application No. 09/284,197, filed on Jun. 25, 1999, now abandoned, filed as 371 of international application No. PCT/US97/19277, filed on Oct. 23, 1997, and which is a continuation-in-part of application No. 08/736,798, filed on Oct. 25, 1996, now Pat. No. 5,918,214, and which is a continuation-in-part of application No. 08/752,136, filed on Nov. 19, 1996, now Pat. No. 6,064,979, and which is a continuation-in-part of application No. 08/826,120, filed on Mar. 27, 1997, now abandoned, and which is a continuation-in-part of application No. 08/854,877, filed on May 12, 1997, now Pat. No. 5,950,173, and which is a continuation-in-part of application No. 08/871,815, filed on Jun. 9, 1997, and which is a continuation-in-part of application No. 08/936,375, filed on Sep. 24, 1997, now abandoned.

Publication Classification

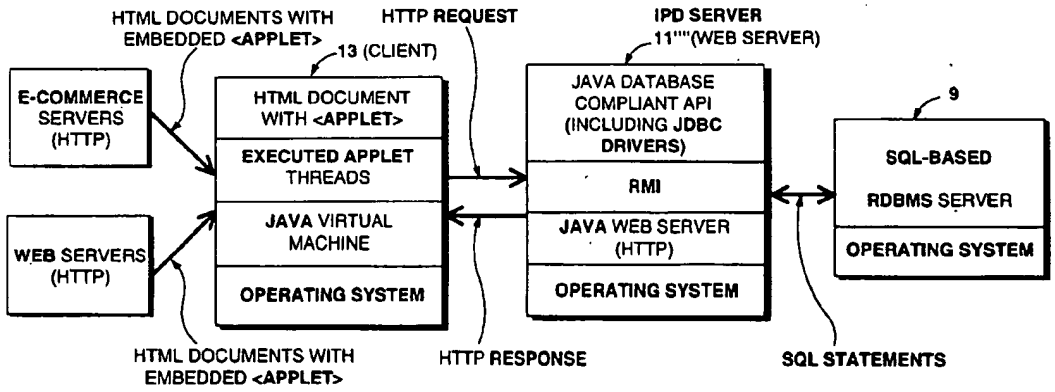
(51) Int. Cl.[7] G06F 17/60

(52) U.S. Cl. 705/26

(57) ABSTRACT

A novel Internet-based consumer product information system and method for use in retail shopping environments, wherein each Web-enabled bar code driven consumer product information kiosk deployed therewithin embodies e-mail messaging capabilities which enable consumers to automatically save and record the URLs of CPI-related Web documents within the message field of a preformatted e-mail message that is transmitted from a retailer-operated e-mail server, to a remote e-mail address specified by the consumer within the retail shopping environment. One or more central e-mail servers are used to collect copies of e-mail documents (and records thereof) transmitted from the Web/e-mail enabled kiosks within each retail shopping environments, for consumer and demographic information analysis, compilation, and storage within RDBMSs that are made accessible to retailers and manufacturers alike for use in product marketing, sales forecasting, customer intelligence, and like operations which enable more effective marketing of consumer products and services in both physical and electronic forms of commerce.

MECHANISM FOR ACCESSING CONSUMER PRODUCT INFORMATION FROM RDBMS SERVER USING CLIENT-SIDE APPLETS EMBEDDED IN HTML DOCUMENTS AND REMOTE METHOD OF INVOCATION (RMI) ON JAVA WEB SERVER



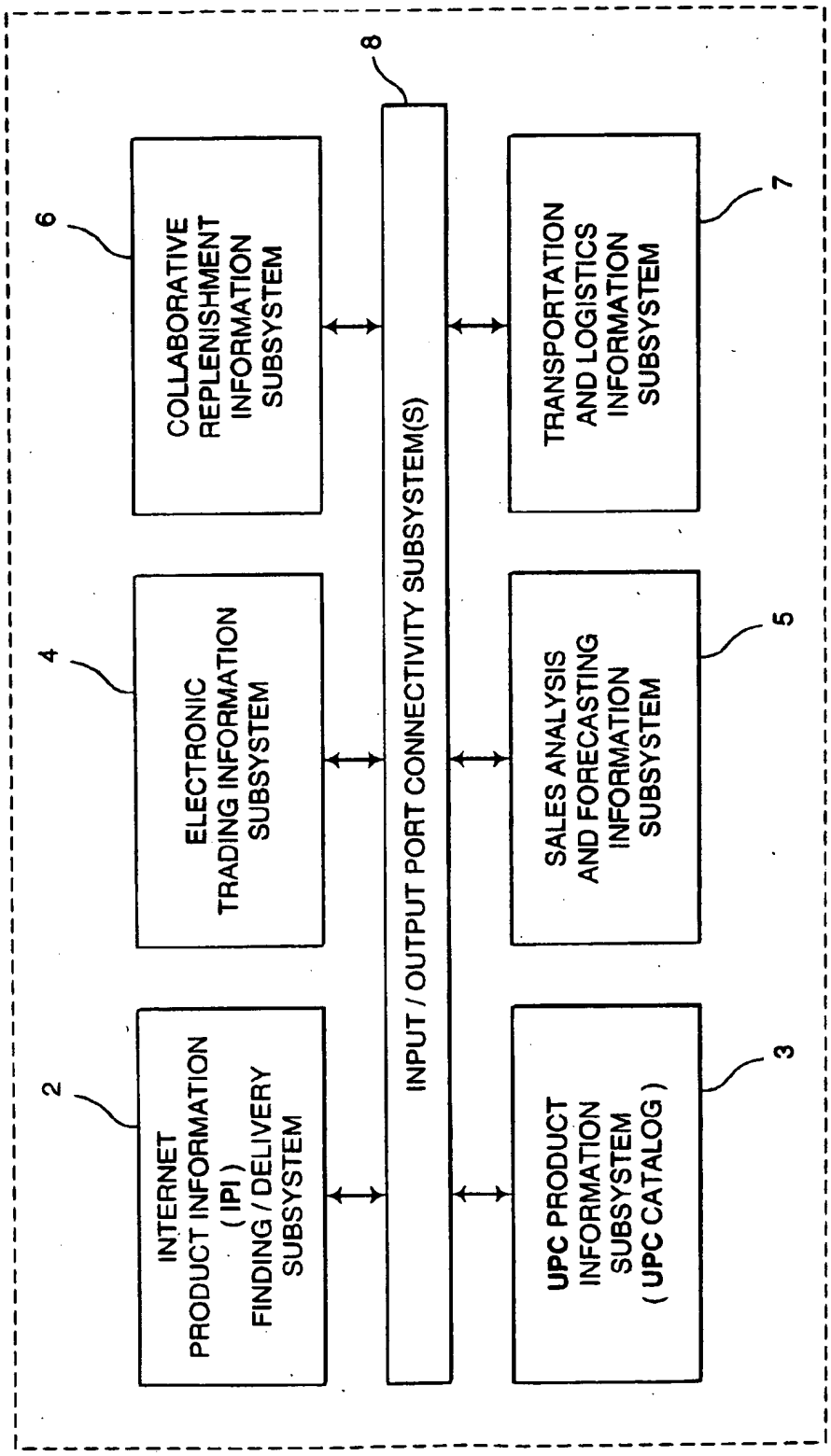

1
2
INTERNET PRODUCT INFORMATION (IPI) FINDING / DELIVERY SUBSYSTEM
3
UPC PRODUCT INFORMATION SUBSYSTEM (UPC CATALOG)
4
ELECTRONIC TRADING INFORMATION SUBSYSTEM
5
SALES ANALYSIS AND FORECASTING INFORMATION SUBSYSTEM
6
COLLABORATIVE REPLENISHMENT INFORMATION SUBSYSTEM
7
TRANSPORTATION AND LOGISTICS INFORMATION SUBSYSTEM
8
INPUT / OUTPUT PORT CONNECTIVITY SUBSYSTEM(S)

FIG. 1

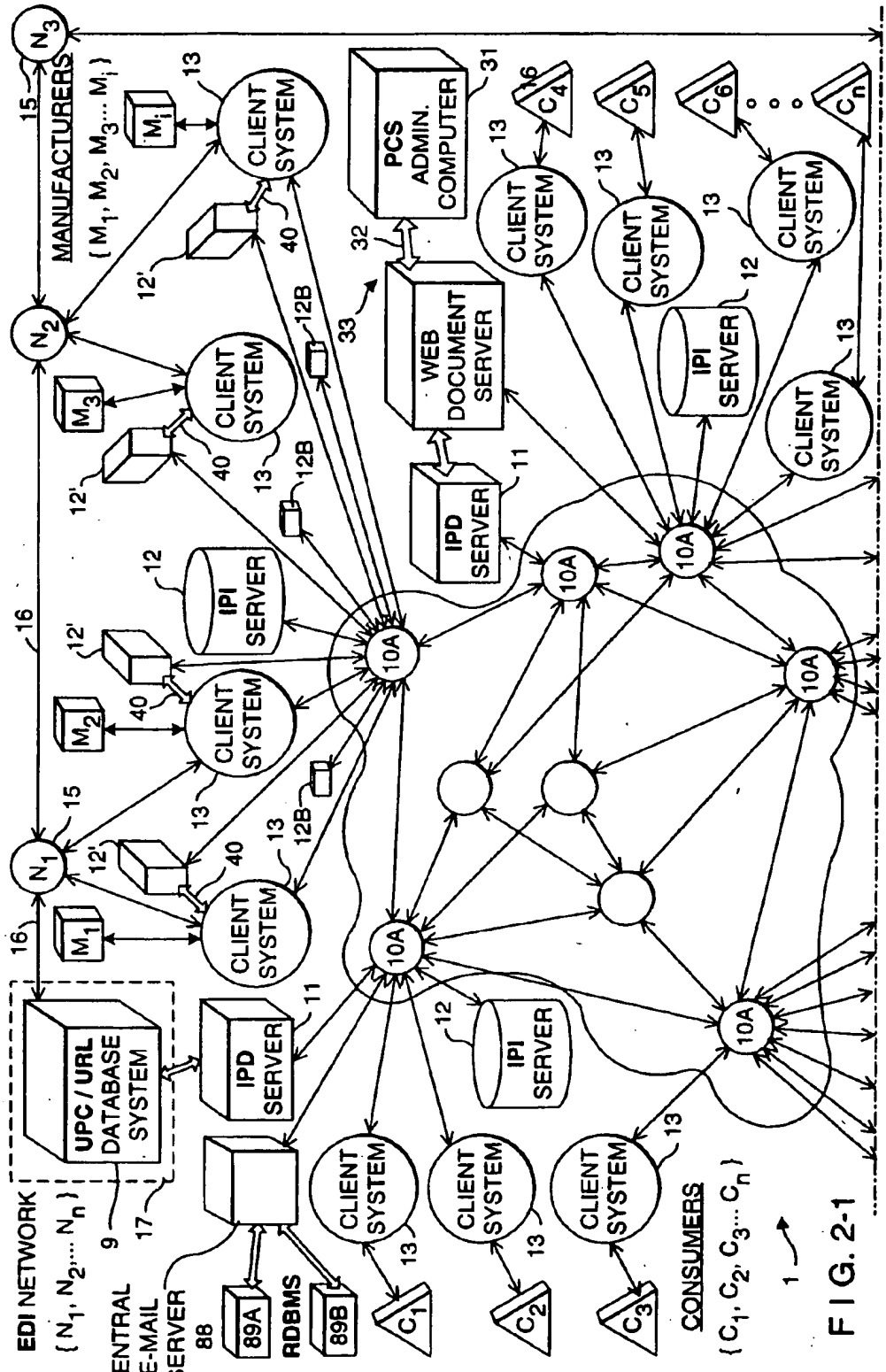

MANUFACTURERS
{M1, M2, M3... Mi}
EDI NETWORK
{N1, N2,... Nn}
CONSUMERS
{C1, C2, C3... Cn}
CLIENT SYSTEM
IPI SERVER
IPD SERVER
WEB DOCUMENT SERVER
PCS ADMIN. COMPUTER
UPC / URL DATABASE SYSTEM
CENTRAL E-MAIL SERVER
RDBMS
10A
12B

FIG. 2-1

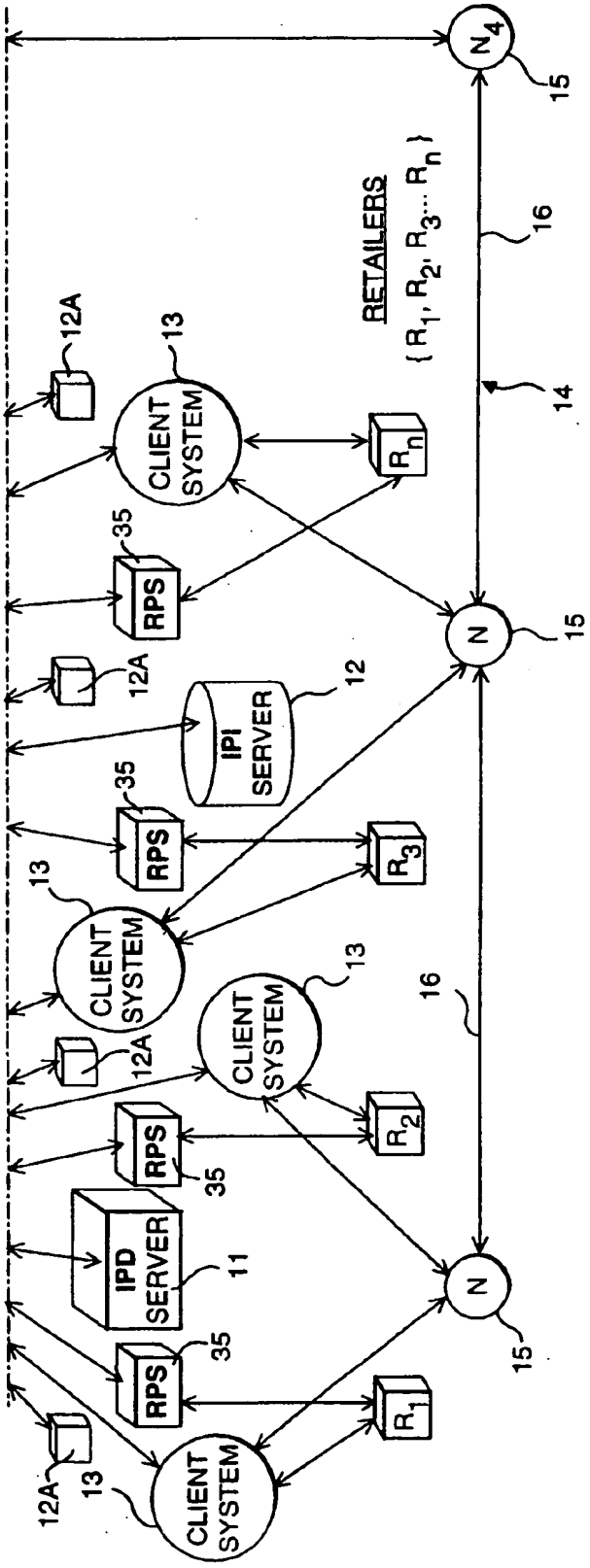

N4
15
16
RETAILERS
{R1, R2, R3... Rn}
14
12A
13
CLIENT SYSTEM
Rn
35
RPS
N
15
12A
IPI SERVER
12
35
RPS
R3
13
CLIENT SYSTEM
13
CLIENT SYSTEM
16
12A
R2
RPS
35
IPD SERVER
11
N
15
35
RPS
R1
12A
13
CLIENT SYSTEM

FIG. 2-2

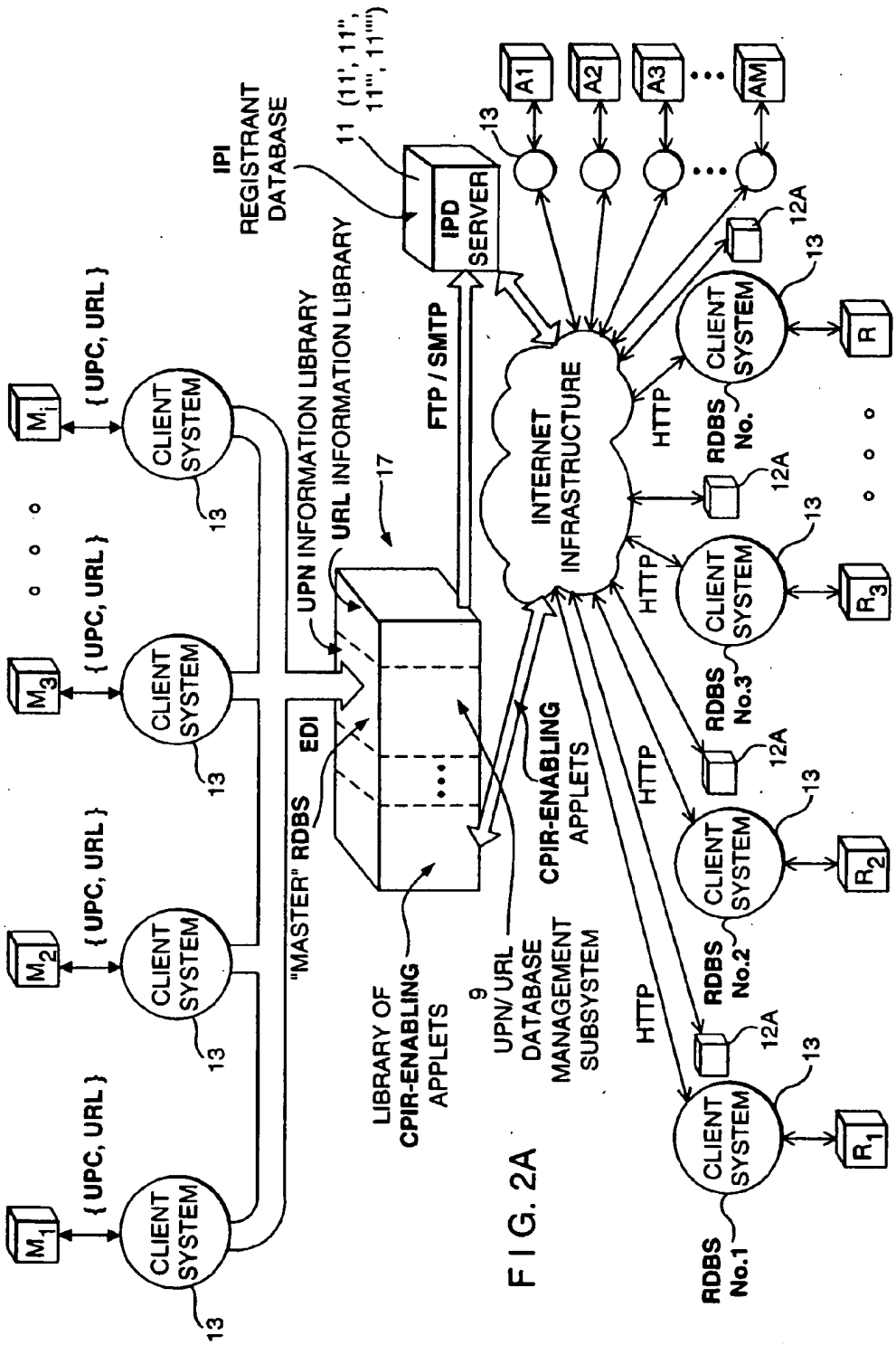

IPI REGISTRANT DATABASE
11 (11', 11", 11"', 11"")
IPD SERVER
13
A1
A2
A3
AM
12A
CLIENT SYSTEM
R
R3
R2
R1
RDBS No.
RDBS No.1
RDBS No.2
RDBS No.3
HTTP
INTERNET INFRASTRUCTURE
FTP / SMTP
17
UPN INFORMATION LIBRARY
URL INFORMATION LIBRARY
"MASTER" RDBS
EDI
CPIR-ENABLING APPLETS
LIBRARY OF CPIR-ENABLING APPLETS
9
UPN/URL DATABASE MANAGEMENT SUBSYSTEM
{UPC, URL}
M1
M2
M3
Mi

FIG. 2A

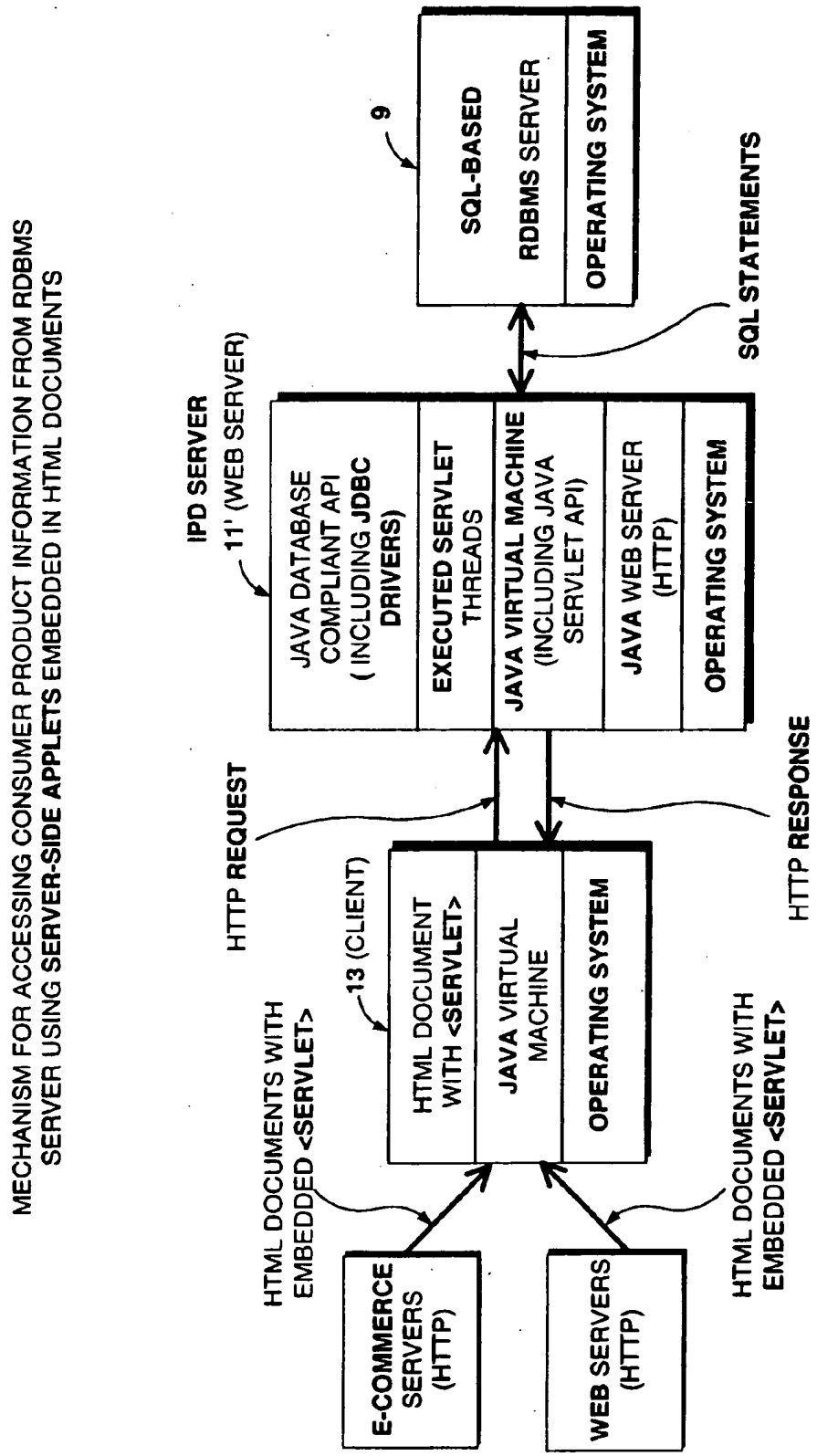

MECHANISM FOR ACCESSING CONSUMER PRODUCT INFORMATION FROM RDBMS SERVER USING SERVER-SIDE APPLETS EMBEDDED IN HTML DOCUMENTS
9
SQL-BASED RDBMS SERVER
OPERATING SYSTEM
SQL STATEMENTS
IPD SERVER
11' (WEB SERVER)
JAVA DATABASE COMPLIANT API (INCLUDING JDBC DRIVERS)
EXECUTED SERVLET THREADS
JAVA VIRTUAL MACHINE (INCLUDING JAVA SERVLET API)
JAVA WEB SERVER (HTTP)
OPERATING SYSTEM
HTTP REQUEST
HTTP RESPONSE
13 (CLIENT)
HTML DOCUMENT WITH <SERVLET>
JAVA VIRTUAL MACHINE
OPERATING SYSTEM
HTML DOCUMENTS WITH EMBEDDED <SERVLET>
HTML DOCUMENTS WITH EMBEDDED <SERVLET>
E-COMMERCE SERVERS (HTTP)
WEB SERVERS (HTTP)

FIG. 2B1

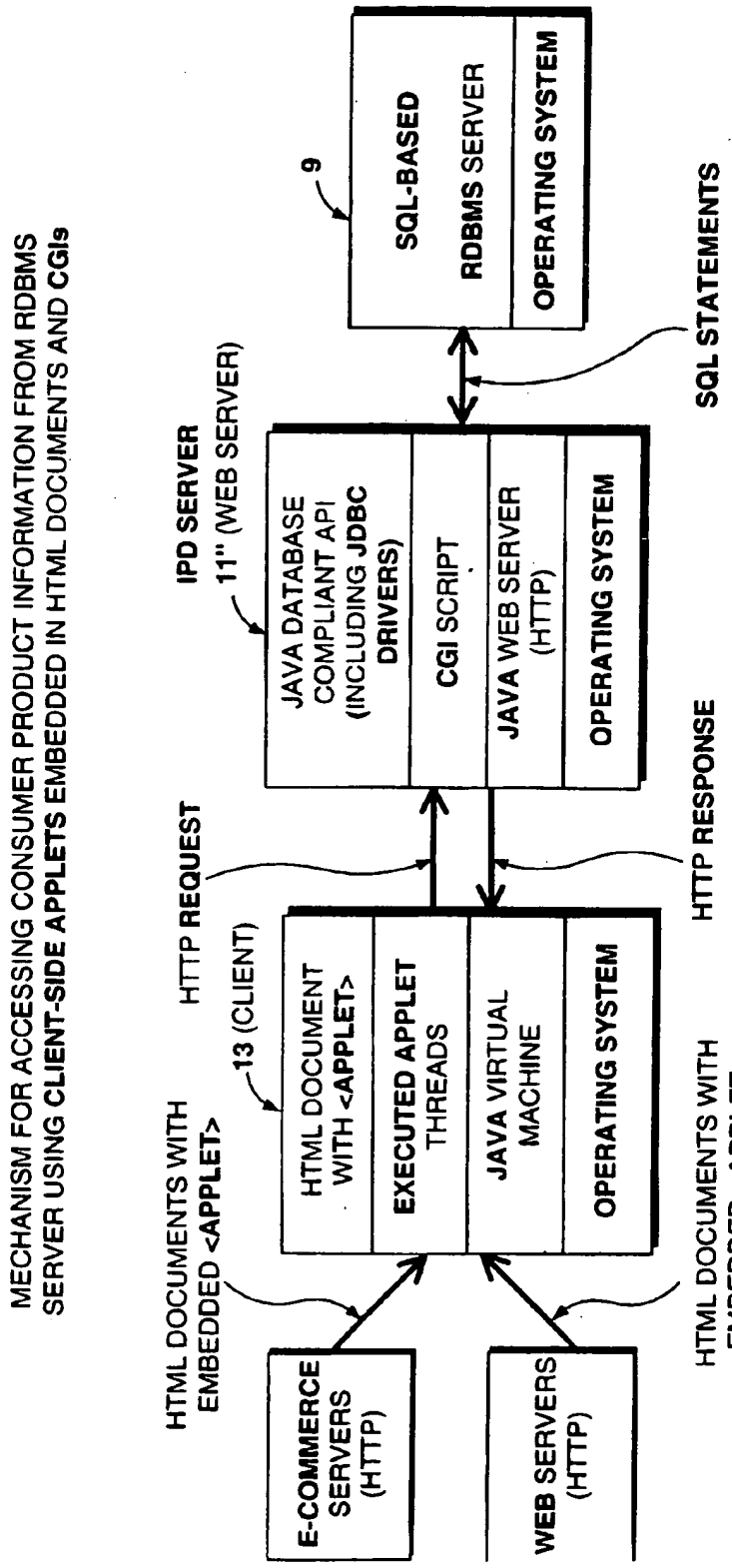

MECHANISM FOR ACCESSING CONSUMER PRODUCT INFORMATION FROM RDBMS SERVER USING CLIENT-SIDE APPLETS EMBEDDED IN HTML DOCUMENTS AND CGIs
9
SQL-BASED RDBMS SERVER
OPERATING SYSTEM
SQL STATEMENTS
IPD SERVER 11" (WEB SERVER)
JAVA DATABASE COMPLIANT API (INCLUDING JDBC DRIVERS)
CGI SCRIPT
JAVA WEB SERVER (HTTP)
OPERATING SYSTEM
HTTP REQUEST
HTTP RESPONSE
13 (CLIENT)
HTML DOCUMENT WITH <APPLET>
EXECUTED APPLET THREADS
JAVA VIRTUAL MACHINE
OPERATING SYSTEM
HTML DOCUMENTS WITH EMBEDDED <APPLET>
HTML DOCUMENTS WITH EMBEDDED <APPLET>
E-COMMERCE SERVERS (HTTP)
WEB SERVERS (HTTP)

FIG. 2B2

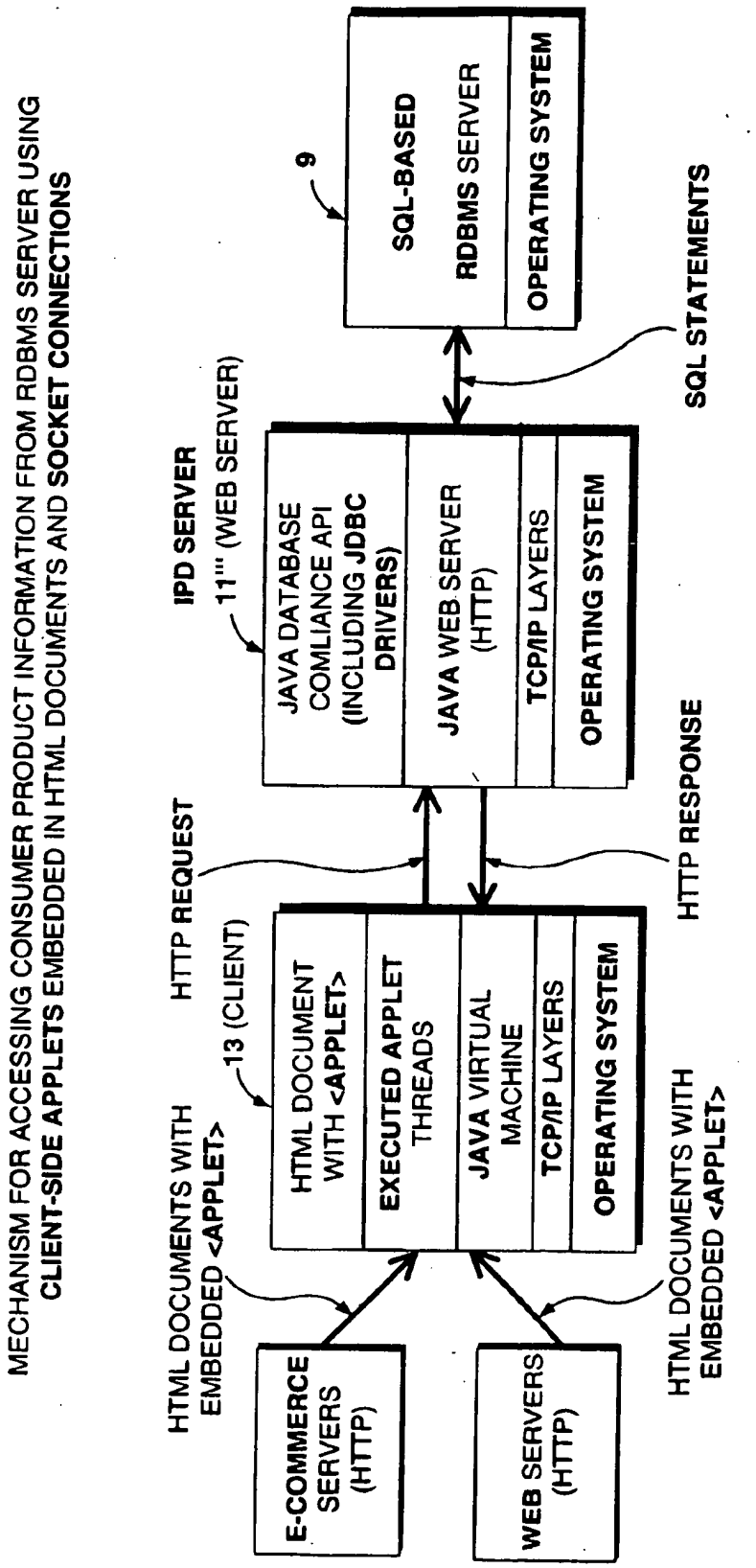

MECHANISM FOR ACCESSING CONSUMER PRODUCT INFORMATION FROM RDBMS SERVER USING CLIENT-SIDE APPLETS EMBEDDED IN HTML DOCUMENTS AND SOCKET CONNECTIONS
9
SQL-BASED RDBMS SERVER
OPERATING SYSTEM
SQL STATEMENTS
IPD SERVER
11''' (WEB SERVER)
JAVA DATABASE COMLIANCE API (INCLUDING JDBC DRIVERS)
JAVA WEB SERVER (HTTP)
TCP/IP LAYERS
OPERATING SYSTEM
HTTP REQUEST
HTTP RESPONSE
13 (CLIENT)
HTML DOCUMENT WITH <APPLET>
EXECUTED APPLET THREADS
JAVA VIRTUAL MACHINE
TCP/IP LAYERS
OPERATING SYSTEM
HTML DOCUMENTS WITH EMBEDDED <APPLET>
HTML DOCUMENTS WITH EMBEDDED <APPLET>
E-COMMERCE SERVERS (HTTP)
WEB SERVERS (HTTP)

FIG. 2B3

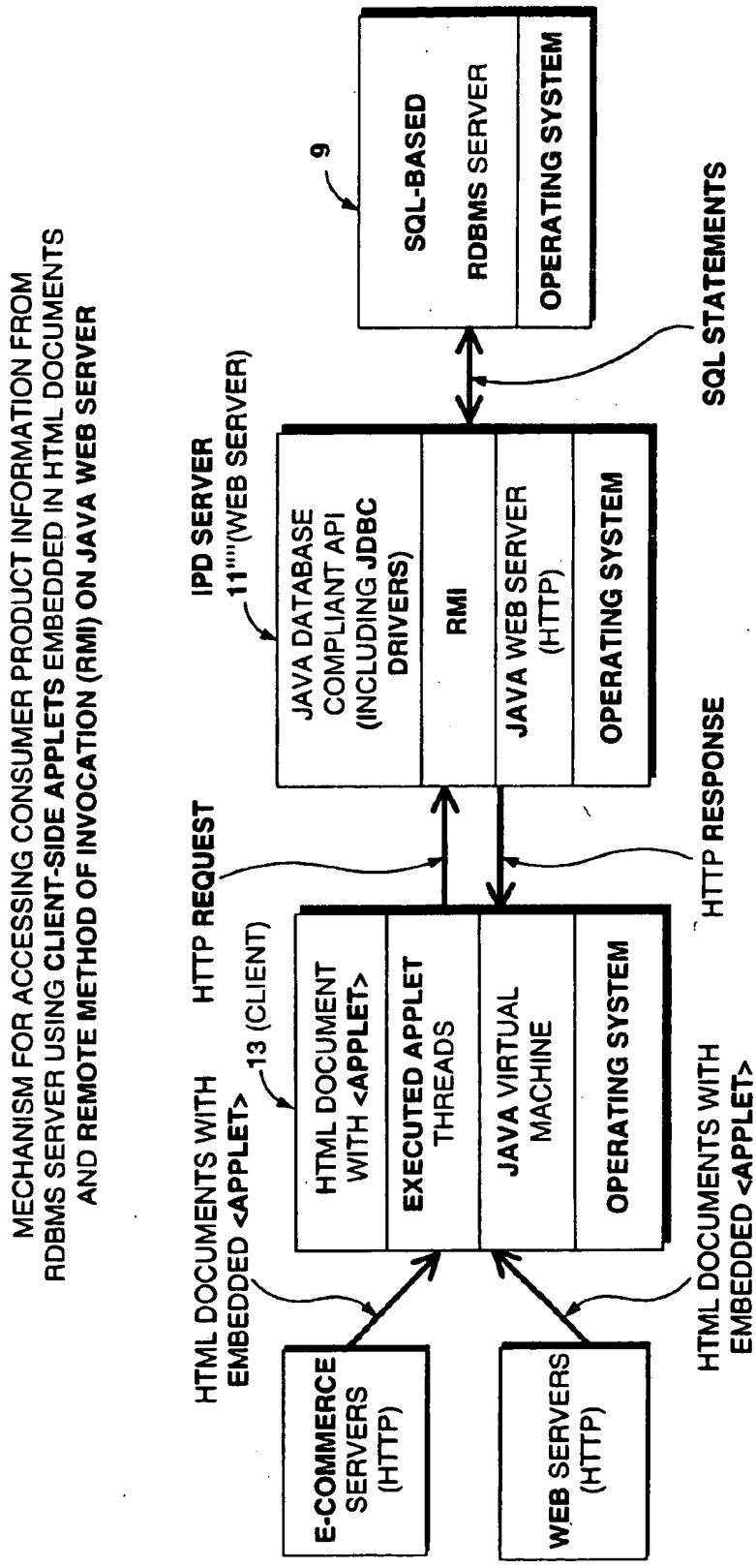

MECHANISM FOR ACCESSING CONSUMER PRODUCT INFORMATION FROM RDBMS SERVER USING CLIENT-SIDE APPLETS EMBEDDED IN HTML DOCUMENTS AND REMOTE METHOD OF INVOCATION (RMI) ON JAVA WEB SERVER
9
SQL-BASED RDBMS SERVER
OPERATING SYSTEM
SQL STATEMENTS
IPD SERVER 11""(WEB SERVER)
JAVA DATABASE COMPLIANT API (INCLUDING JDBC DRIVERS)
RMI
JAVA WEB SERVER (HTTP)
OPERATING SYSTEM
HTTP REQUEST
HTTP RESPONSE
13 (CLIENT)
HTML DOCUMENT WITH <APPLET>
EXECUTED APPLET THREADS
JAVA VIRTUAL MACHINE
OPERATING SYSTEM
HTML DOCUMENTS WITH EMBEDDED <APPLET>
HTML DOCUMENTS WITH EMBEDDED <APPLET>
E-COMMERCE SERVERS (HTTP)
WEB SERVERS (HTTP)

FIG. 2B4

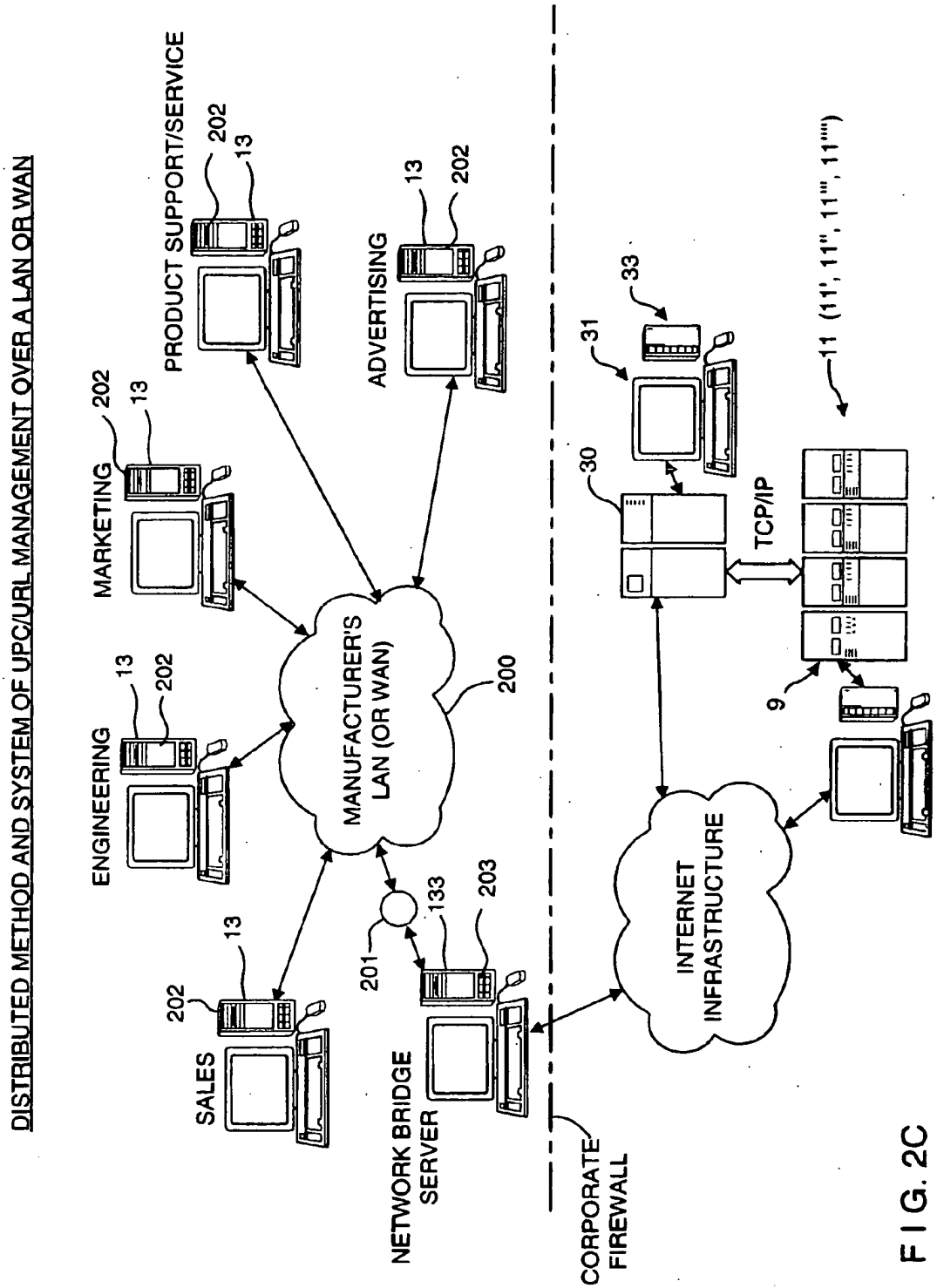

DISTRIBUTED METHOD AND SYSTEM OF UPC/URL MANAGEMENT OVER A LAN OR WAN
PRODUCT SUPPORT/SERVICE
202
13
ADVERTISING
13
202
MARKETING
202
13
ENGINEERING
13
202
SALES
13
202
MANUFACTURER'S LAN (OR WAN)
200
201
133
203
NETWORK BRIDGE SERVER
CORPORATE FIREWALL
INTERNET INFRASTRUCTURE
33
31
30
TCP/IP
11 (11', 11", 11"', 11"")
9

FIG. 2C

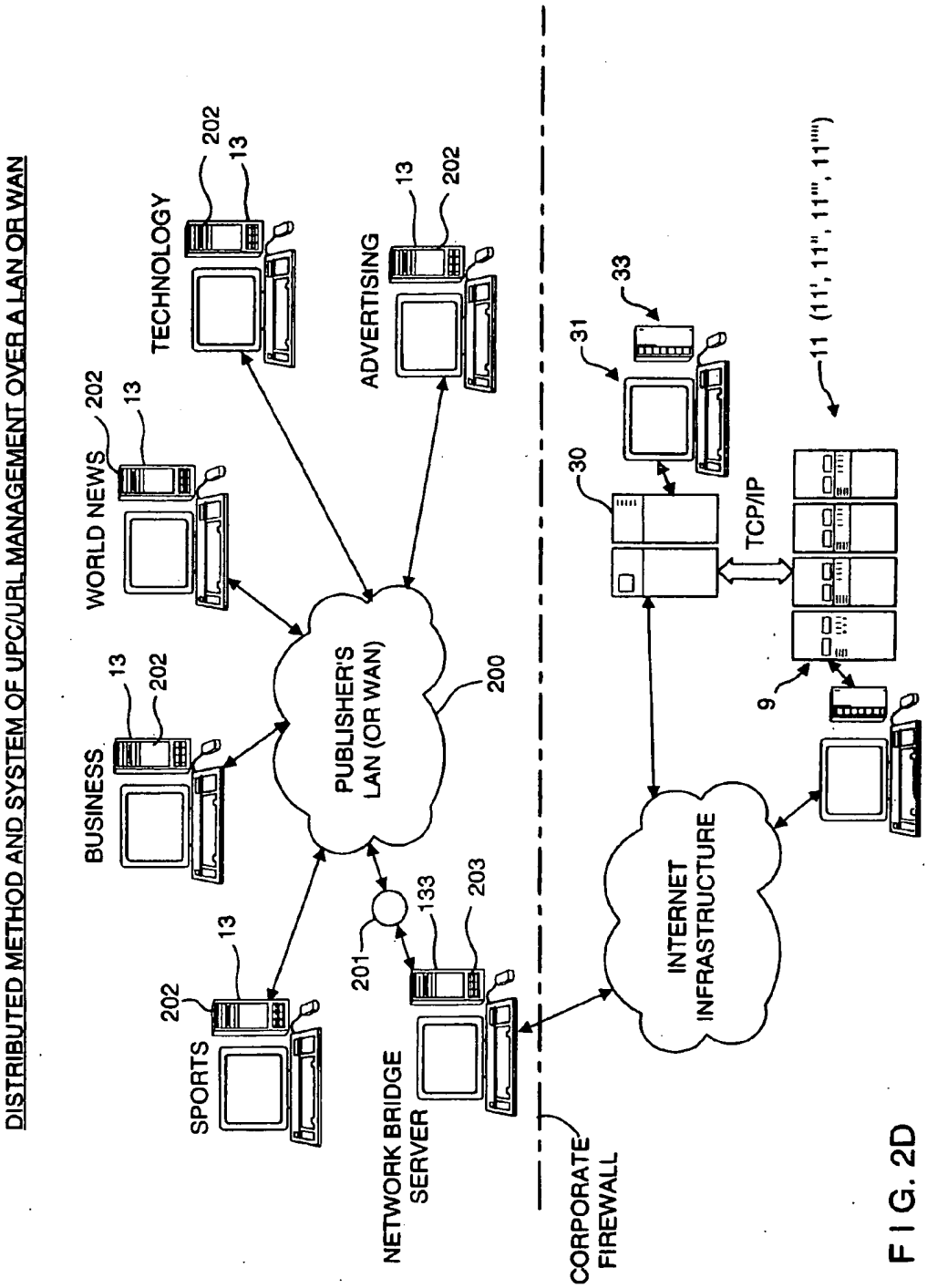

DISTRIBUTED METHOD AND SYSTEM OF UPC/URL MANAGEMENT OVER A LAN OR WAN
TECHNOLOGY
ADVERTISING
WORLD NEWS
BUSINESS
SPORTS
PUBLISHER'S LAN (OR WAN)
200
NETWORK BRIDGE SERVER
133
201
203
CORPORATE FIREWALL
INTERNET INFRASTRUCTURE
TCP/IP
30
31
33
11 (11', 11", 11"', 11"")
9
202
13

FIG. 2D

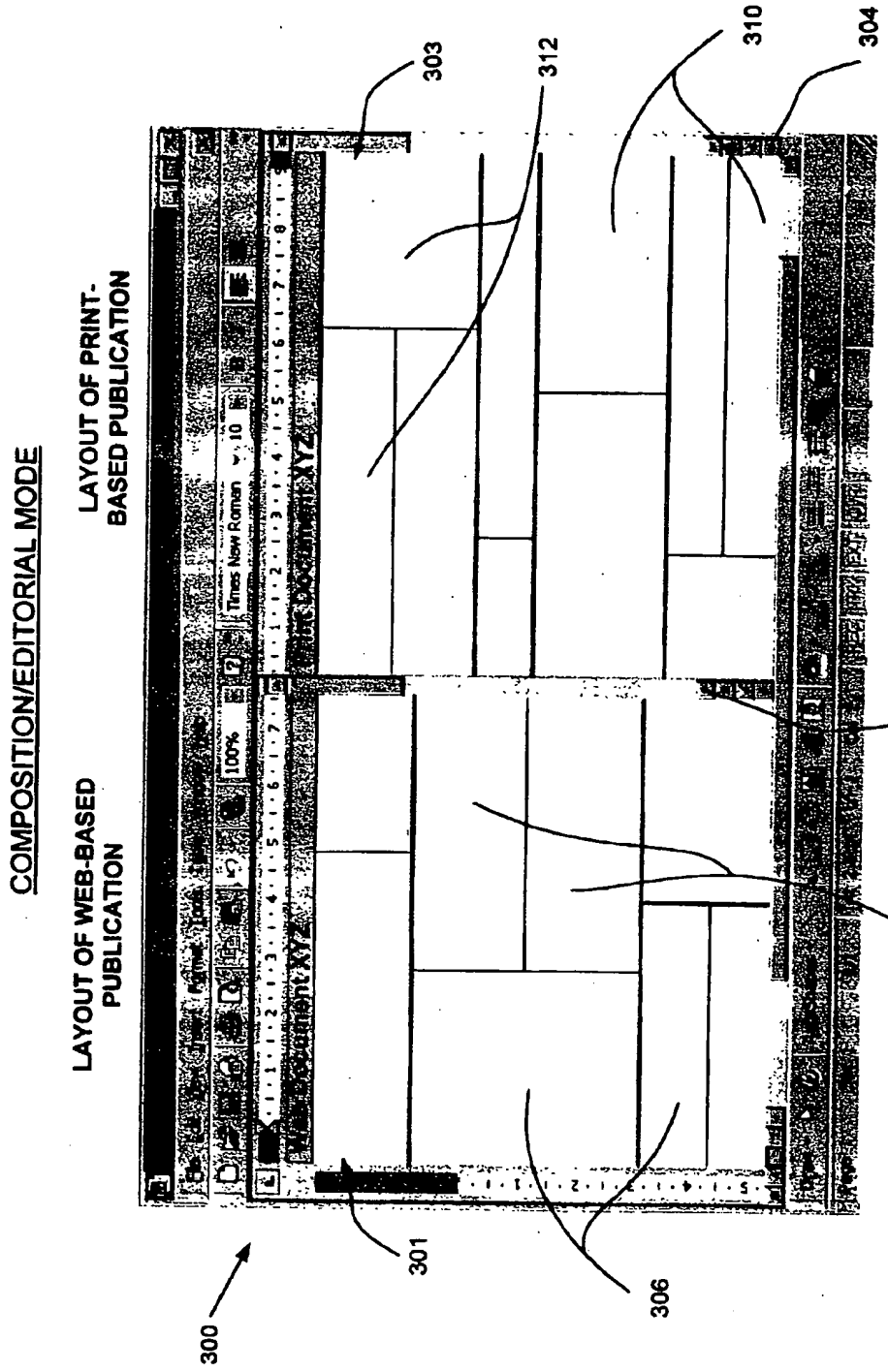

COMPOSITION/EDITORIAL MODE
LAYOUT OF WEB-BASED PUBLICATION
LAYOUT OF PRINT-BASED PUBLICATION
303
312
310
304
301
308
301
306
300

FIG. 2E1

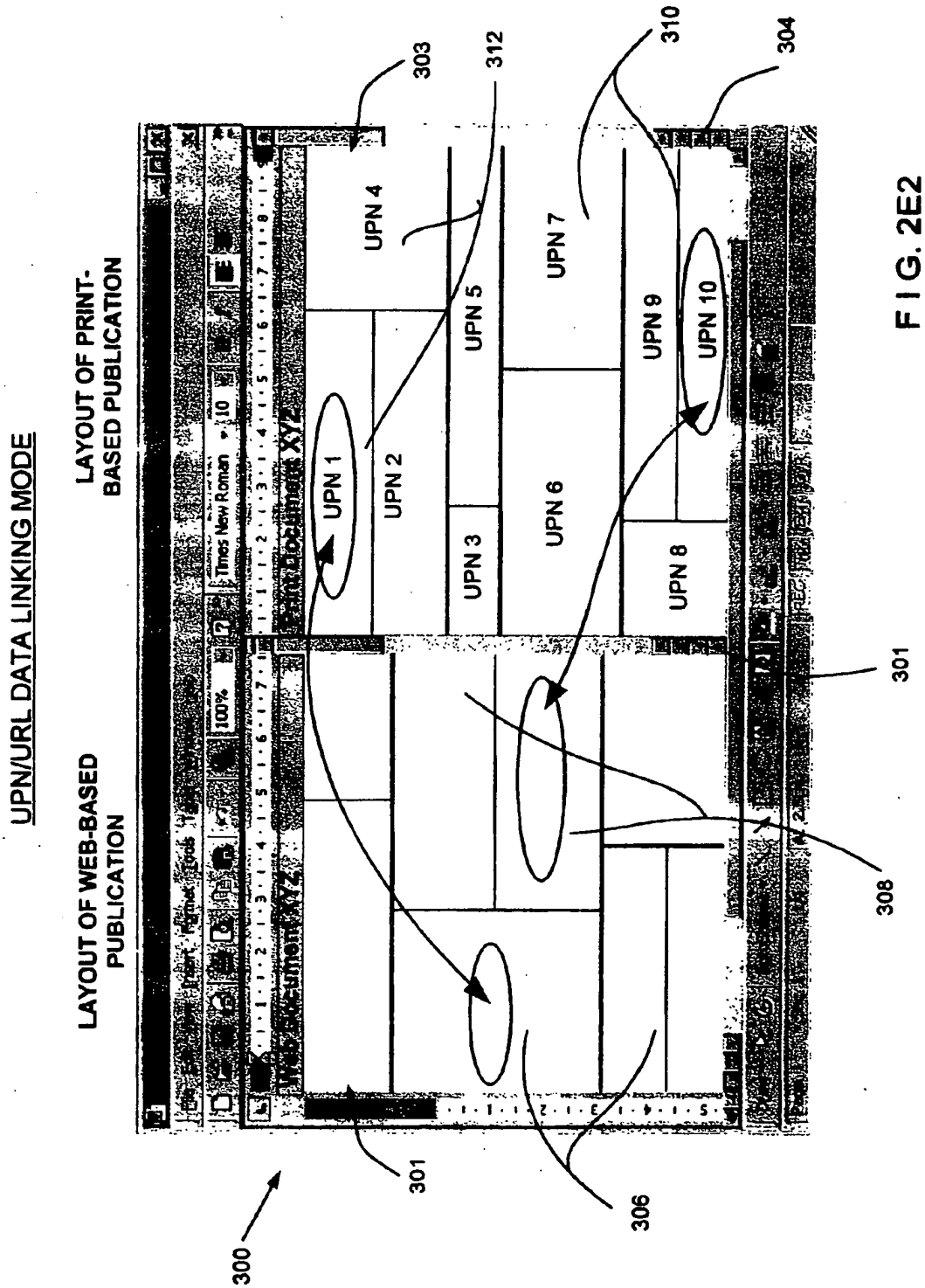

UPN/URL DATA LINKING MODE
LAYOUT OF WEB-BASED PUBLICATION
LAYOUT OF PRINT-BASED PUBLICATION
UPN 1
UPN 2
UPN 3
UPN 4
UPN 5
UPN 6
UPN 7
UPN 8
UPN 9
UPN 10
300
301
303
304
306
308
310
312

FIG. 2E2

UPN/URL LINKS	
UPN	**URL**
UPN 1	URL 1
UPN 2	URL 2
UPN 3	URL 3
UPN 4	URL 4
UPN 5	URL 5
UPN 6	URL 6
UPN 7	URL 7
UPN 8	URL 8
UPN 9	URL 9
UPN 10	URL 10
UPN 11	URL 11
UPN 12	URL 12
UPN 13	URL 13
⋮	⋮
UPN N	URL N
UPN N-1	URL N-1

400

FIG. 2E3

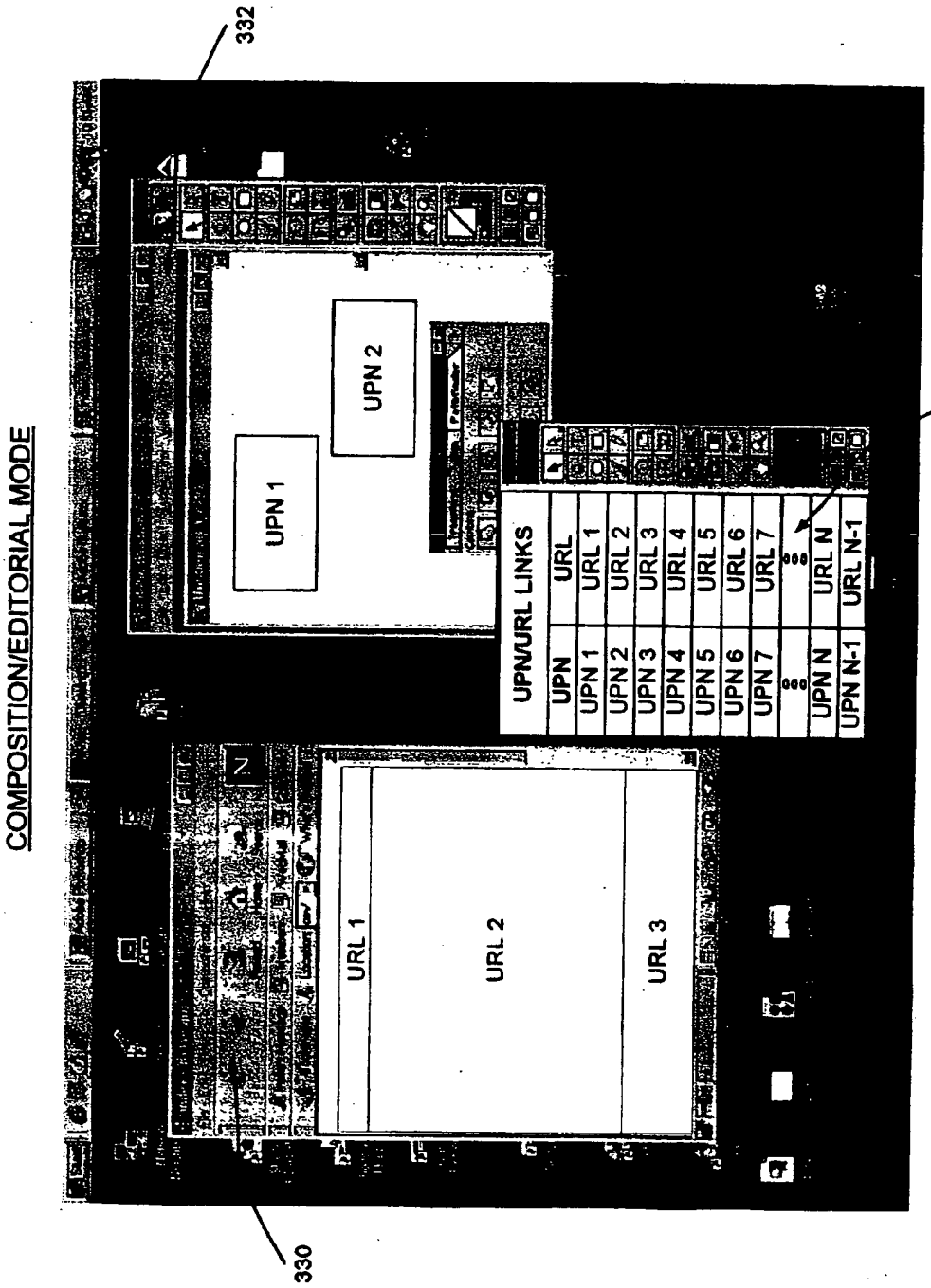

COMPOSITION/EDITORIAL MODE
332
330
400
UPN 1
UPN 2
URL 1
URL 2
URL 3
UPN/URL LINKS
UPN
URL
UPN 1
URL 1
UPN 2
URL 2
UPN 3
URL 3
UPN 4
URL 4
UPN 5
URL 5
UPN 6
URL 6
UPN 7
URL 7
UPN N
URL N
UPN N-1
URL N-1

FIG. 2E4

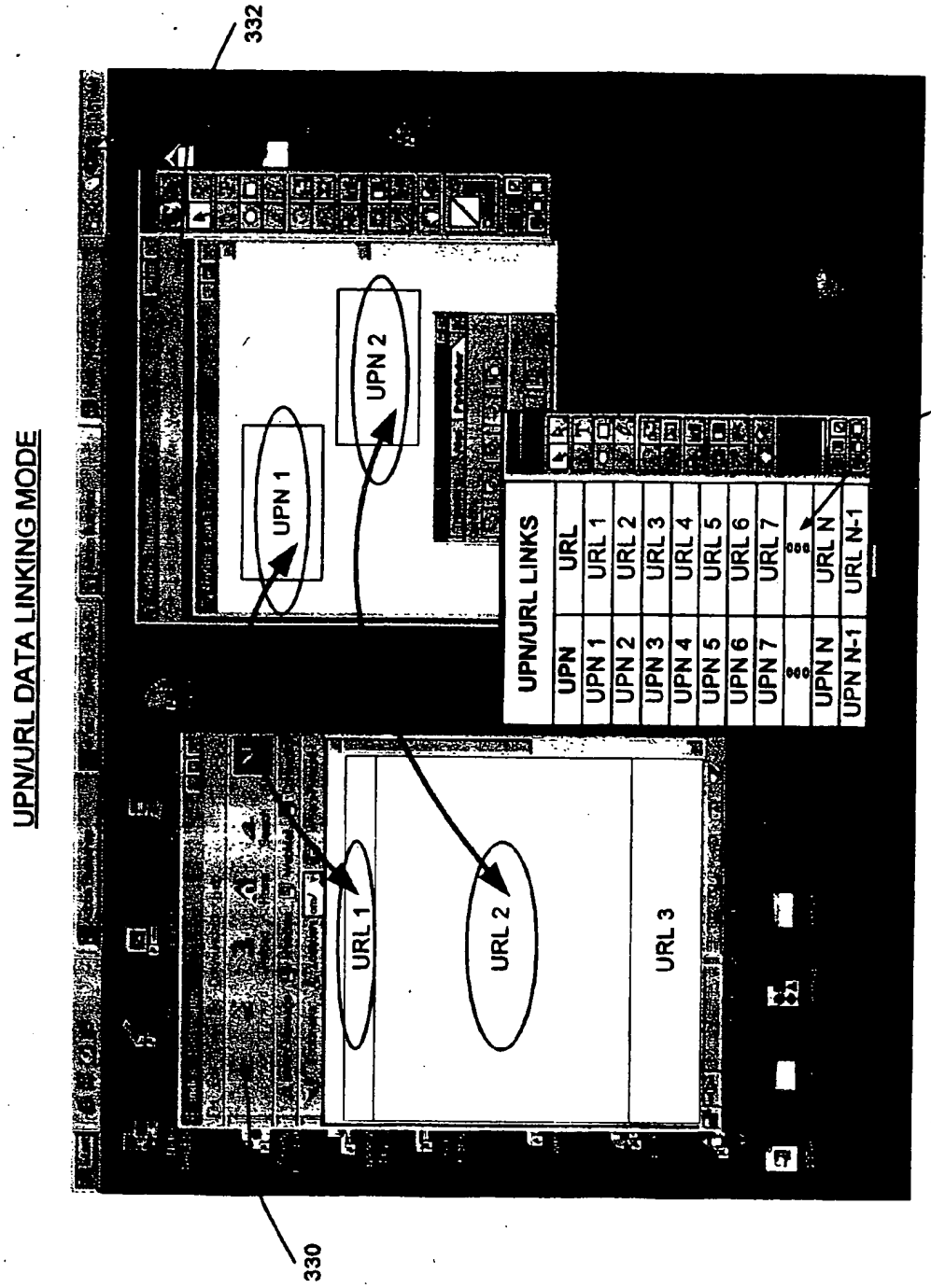

332
UPN/URL DATA LINKING MODE
UPN 1
UPN 2
URL 1
URL 2
URL 3
UPN/URL LINKS
UPN
URL
UPN 1
URL 1
UPN 2
URL 2
UPN 3
URL 3
UPN 4
URL 4
UPN 5
URL 5
UPN 6
URL 6
UPN 7
URL 7
UPN N
URL N
UPN N-1
URL N-1
400
330

FIG. 2E5

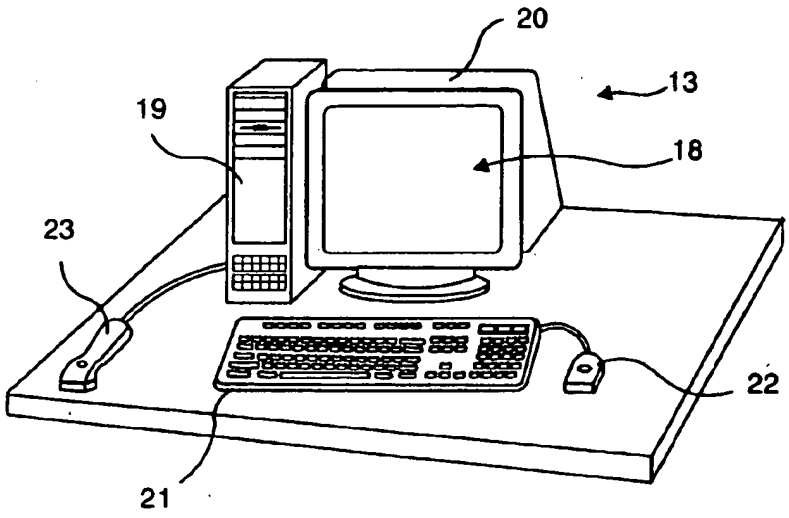


FIG. 3A1

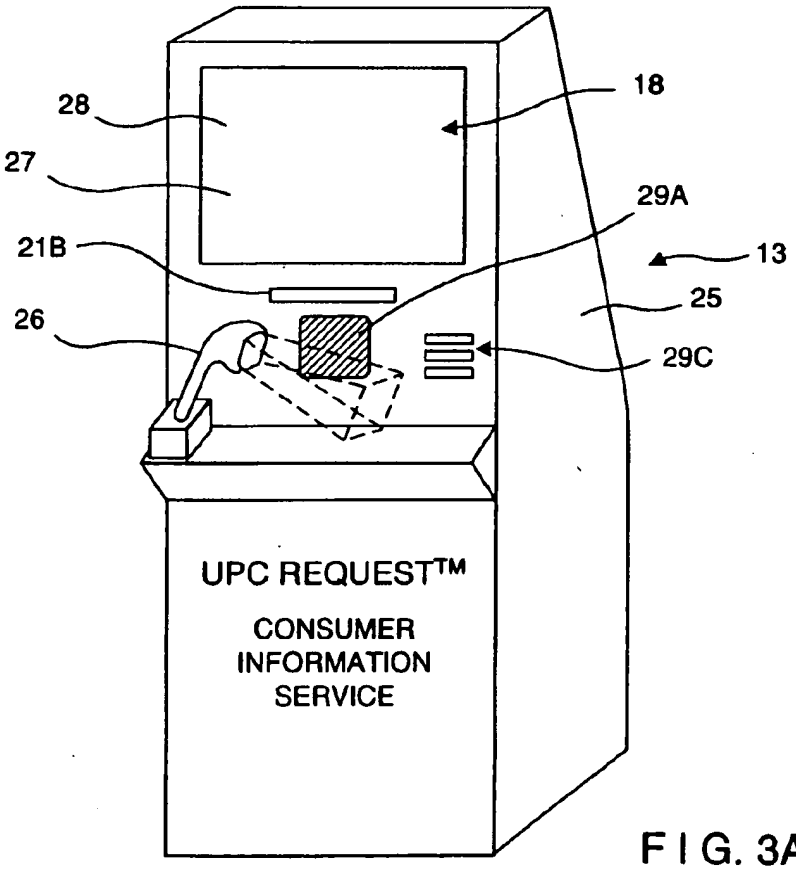
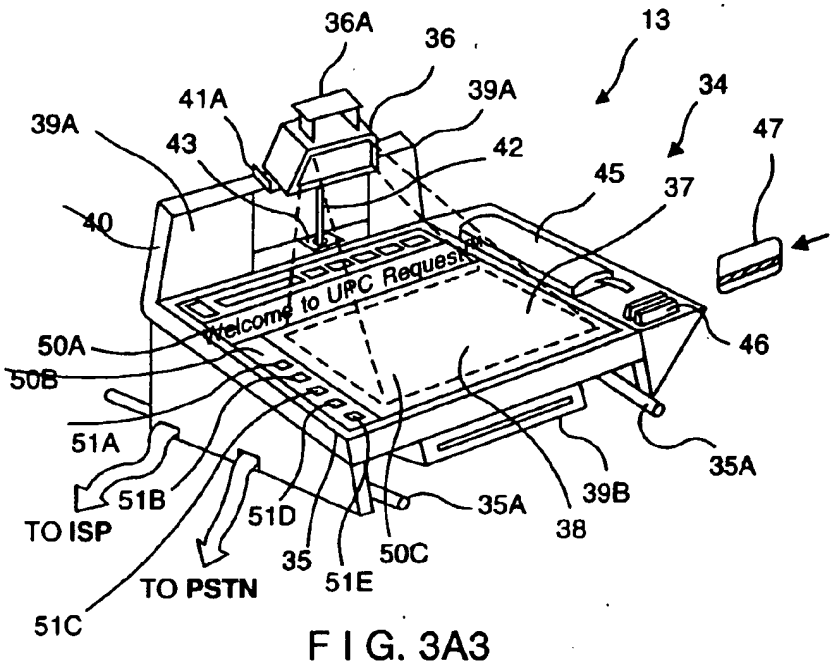
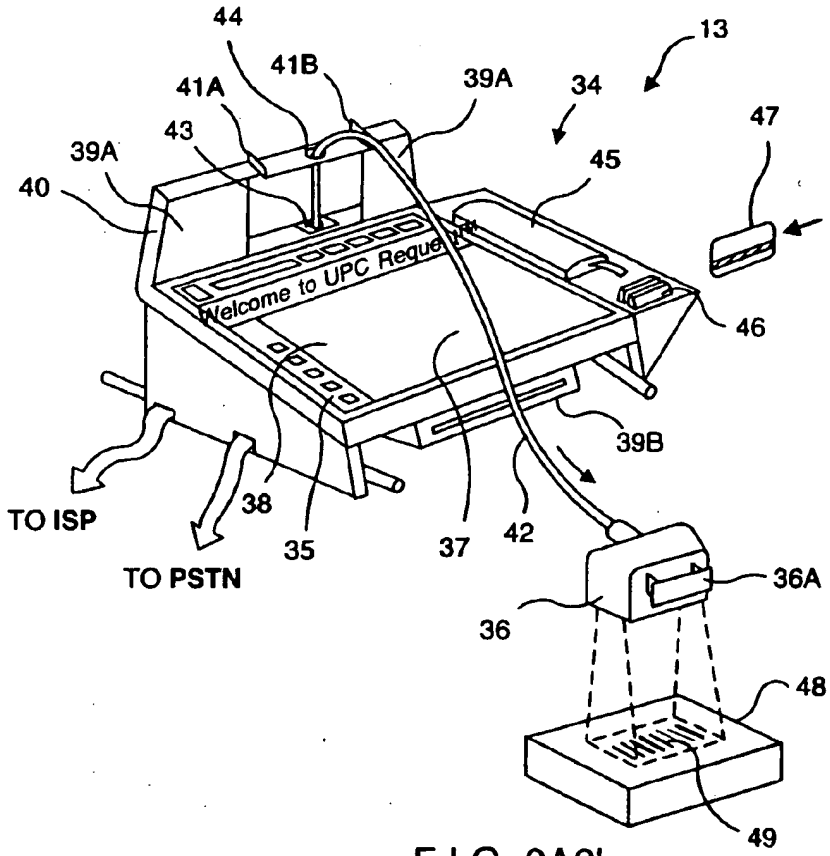


FIG. 3A2



F I G. 3A3



F I G. 3A3'

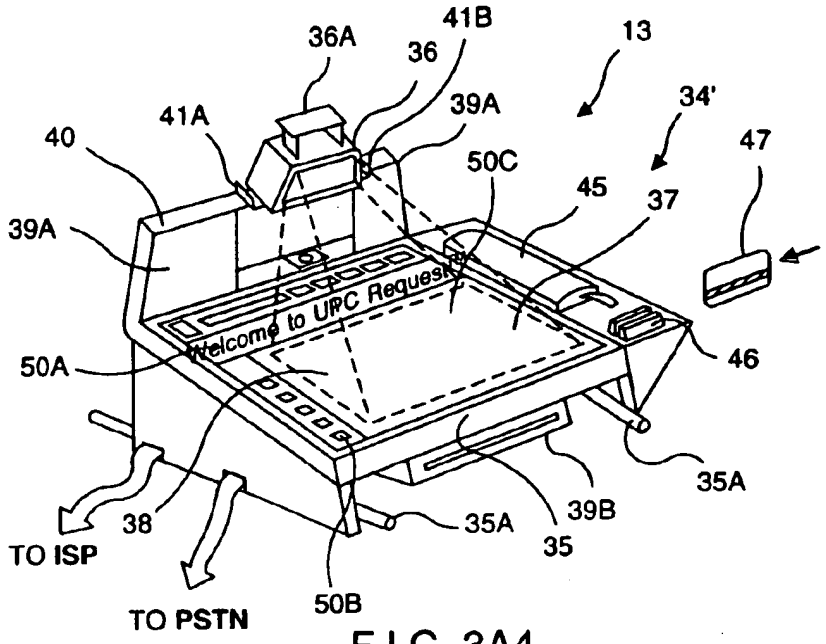


FIG. 3A4

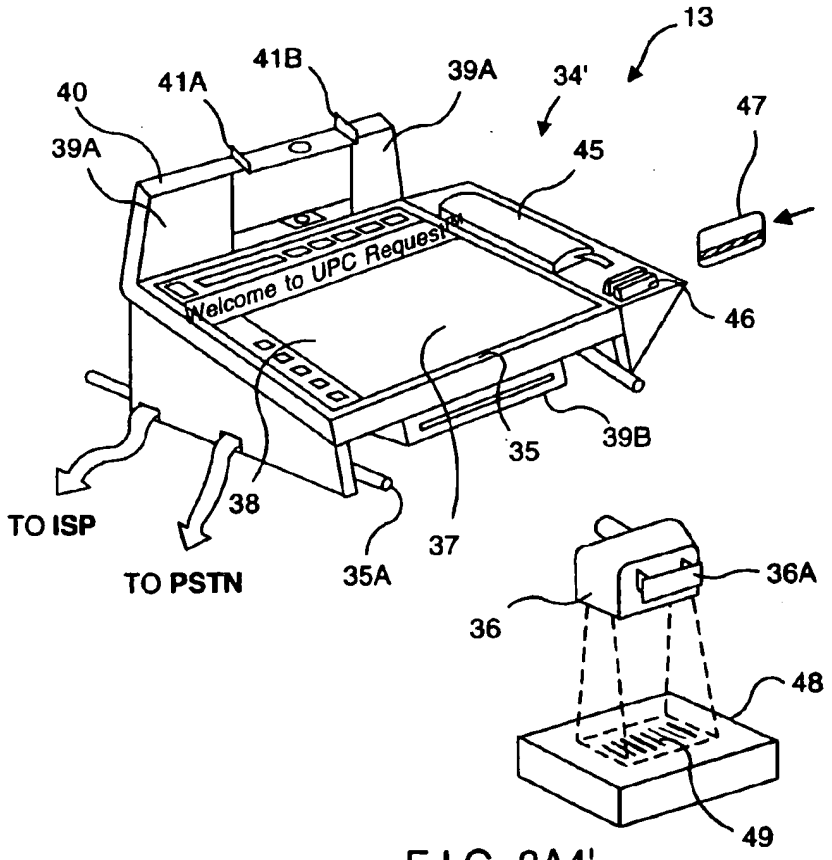


FIG. 3A4'

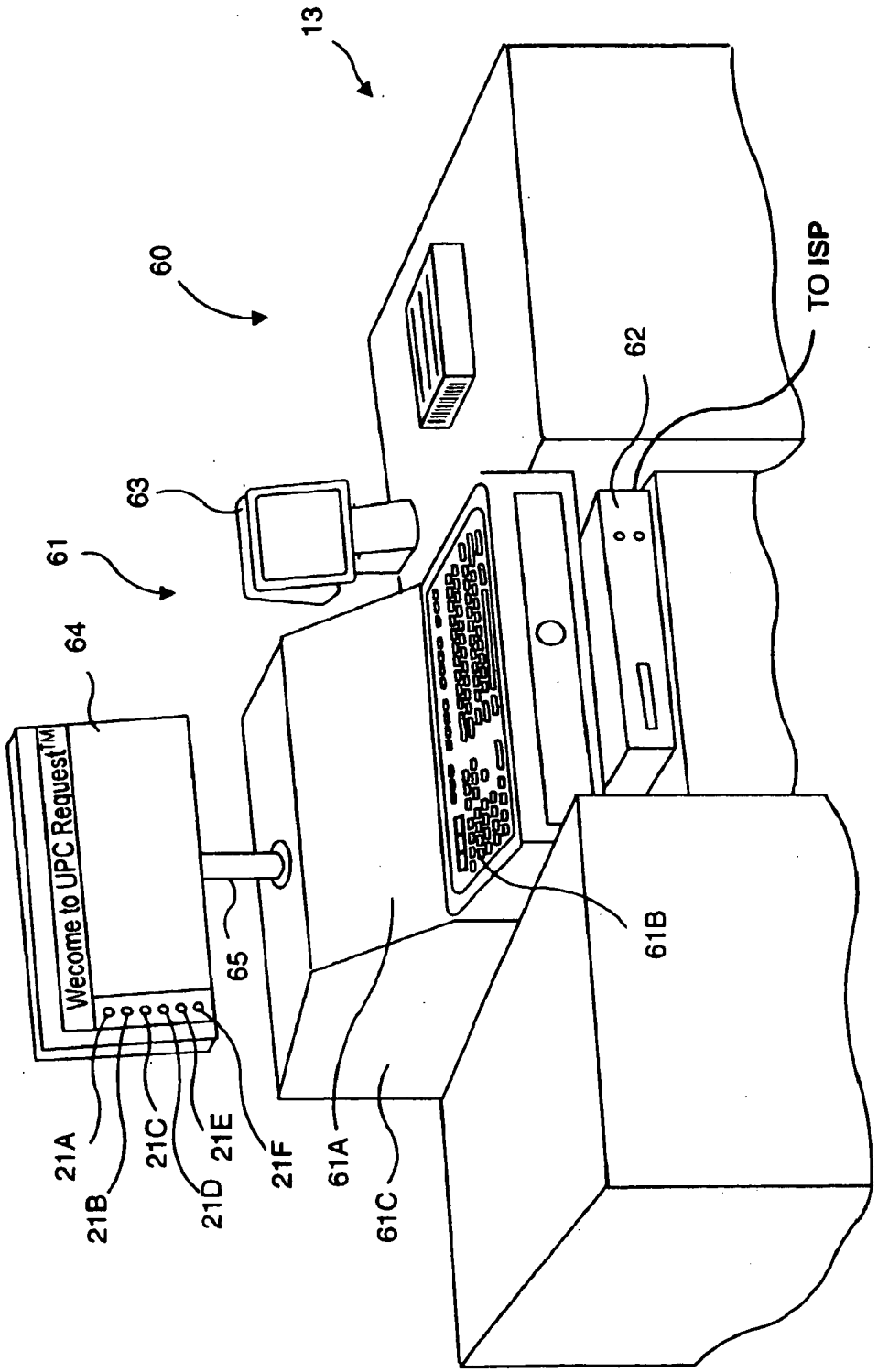

13
60
TO ISP
62
63
61
64
Welcome to UPC Request™
65
61B
21A
21B
21C
21D
21E
21F
61A
61C

FIG. 3A5

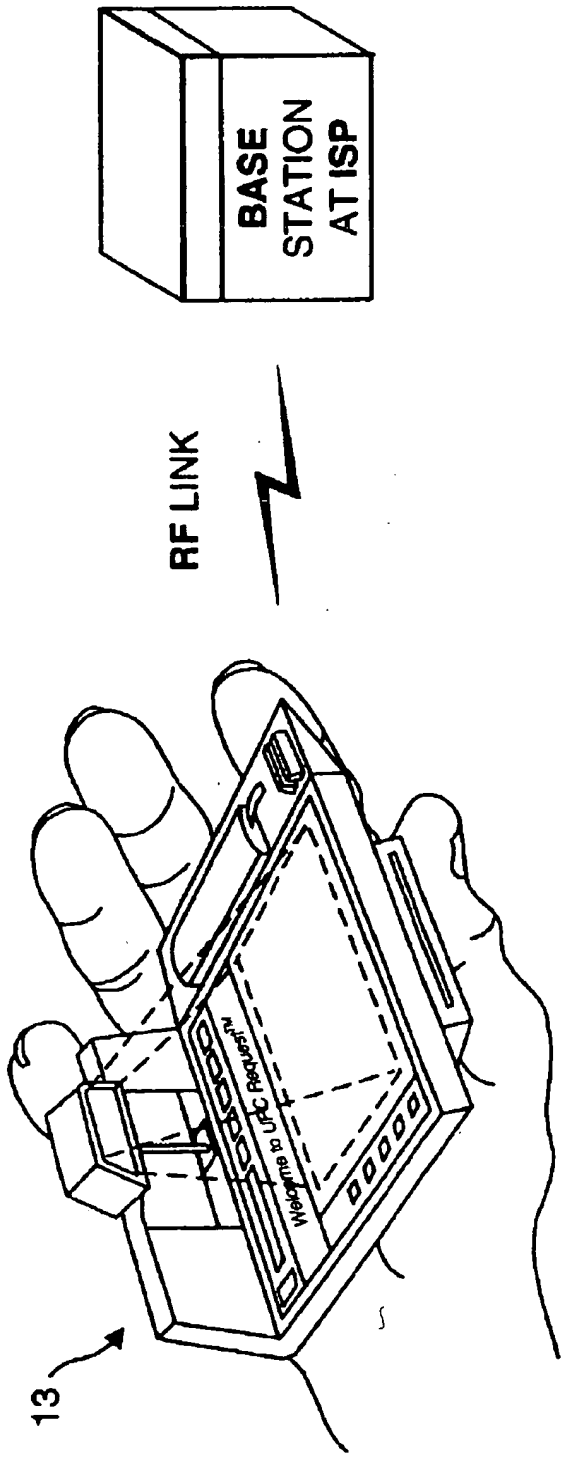


FIG. 3A6

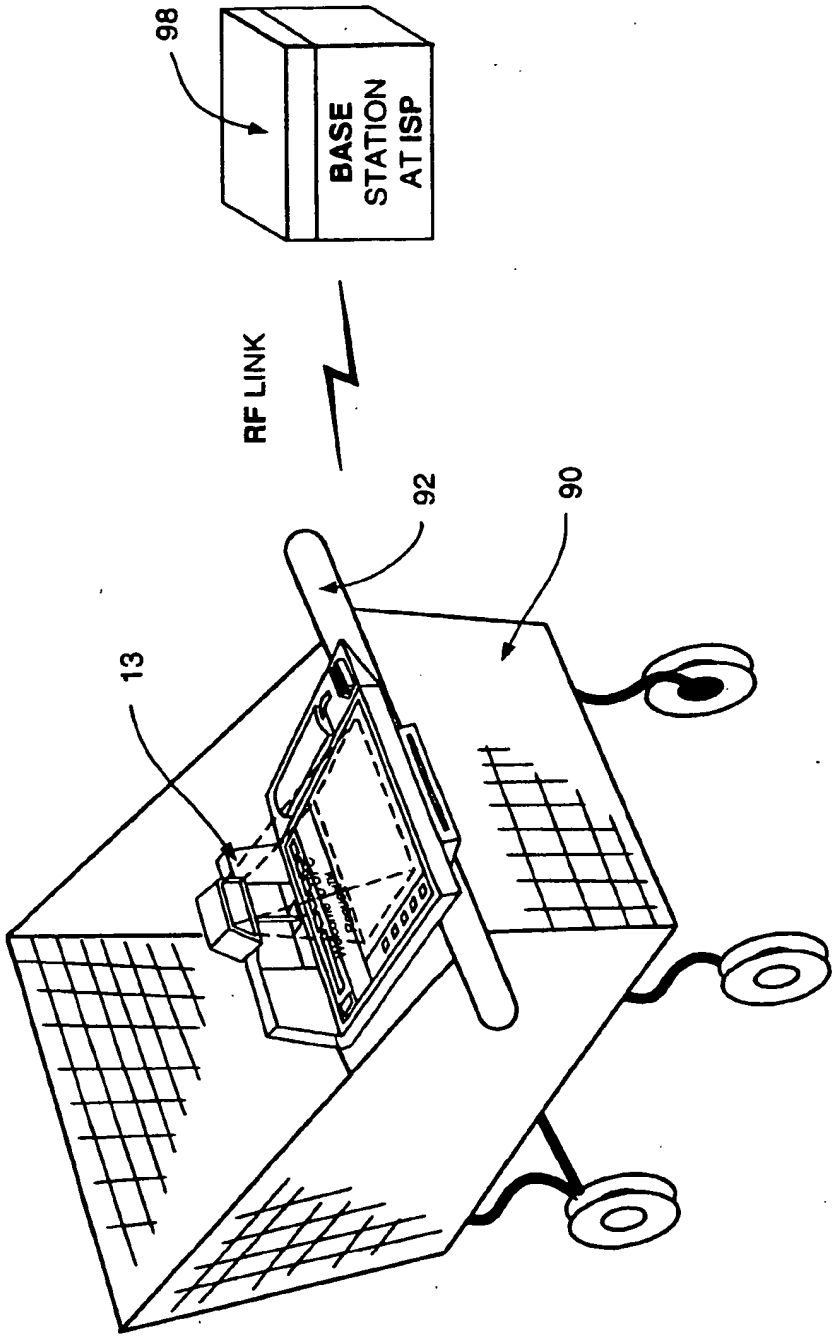


FIG. 3A7

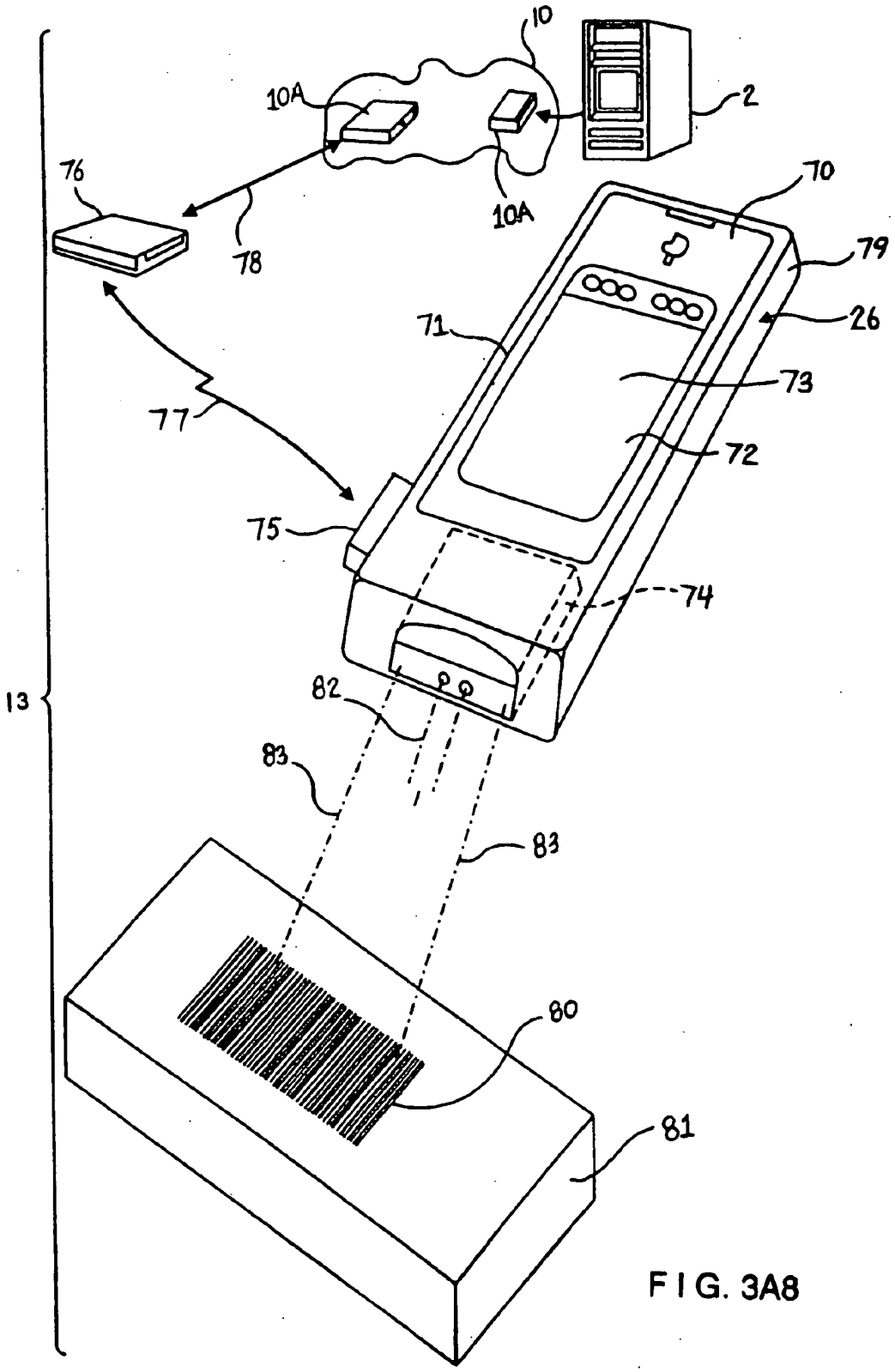


FIG. 3A8

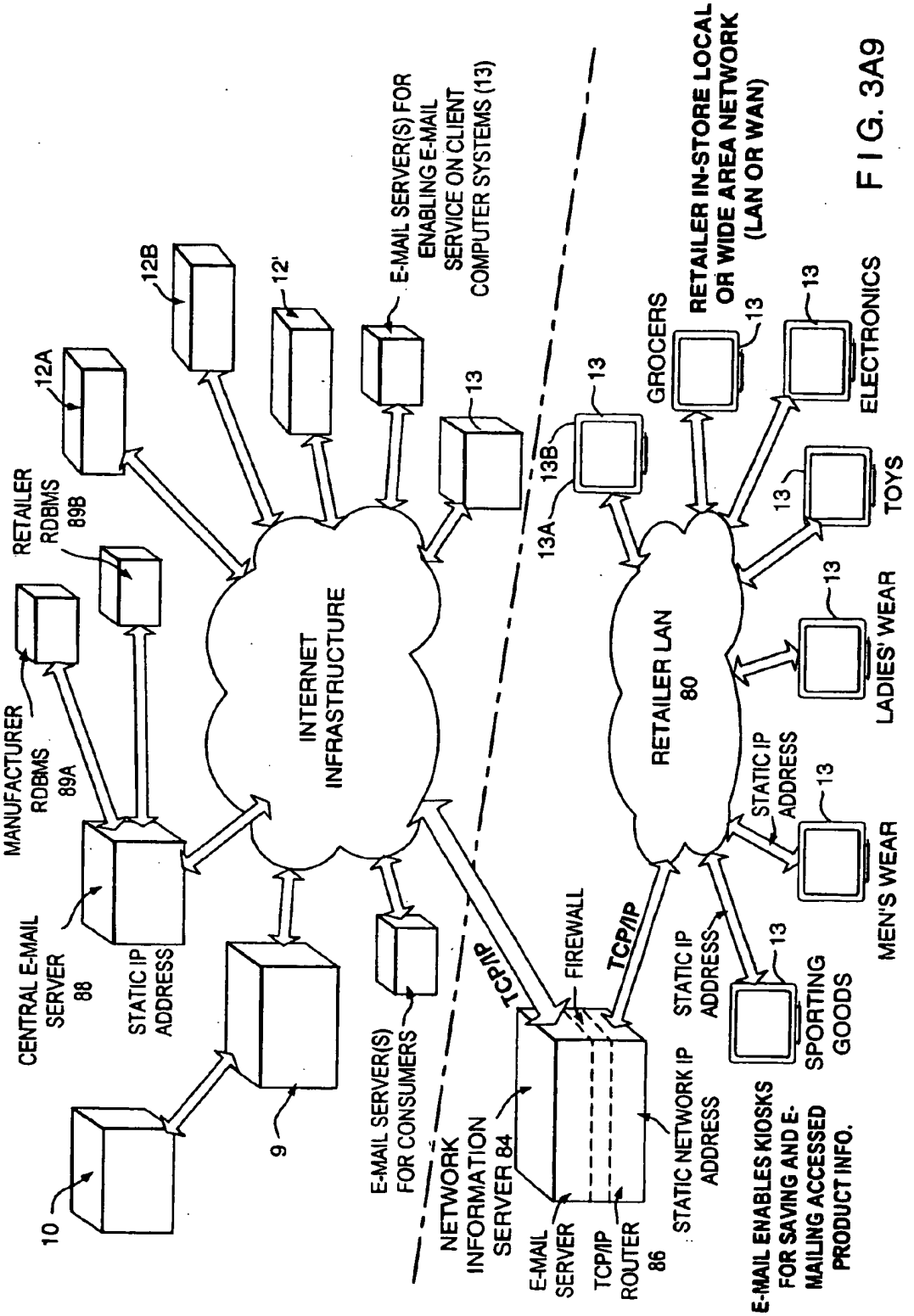

12B
12'
E-MAIL SERVER(S) FOR ENABLING E-MAIL SERVICE ON CLIENT COMPUTER SYSTEMS (13)
RETAILER IN-STORE LOCAL OR WIDE AREA NETWORK (LAN OR WAN)
12A
RETAILER RDBMS 89B
INTERNET INFRASTRUCTURE
13
13A
13B
GROCERS
ELECTRONICS
TOYS
LADIES' WEAR
MEN'S WEAR
SPORTING GOODS
RETAILER LAN 80
STATIC IP ADDRESS
MANUFACTURER RDBMS 89A
CENTRAL E-MAIL SERVER 88
STATIC IP ADDRESS
E-MAIL SERVER(S) FOR CONSUMERS
NETWORK INFORMATION SERVER 84
TCP/IP
FIREWALL
TCP/IP
E-MAIL SERVER
TCP/IP ROUTER 86
STATIC NETWORK IP ADDRESS
STATIC IP ADDRESS
E-MAIL ENABLES KIOSKS FOR SAVING AND E-MAILING ACCESSED PRODUCT INFO.
10
9

FIG. 3A9

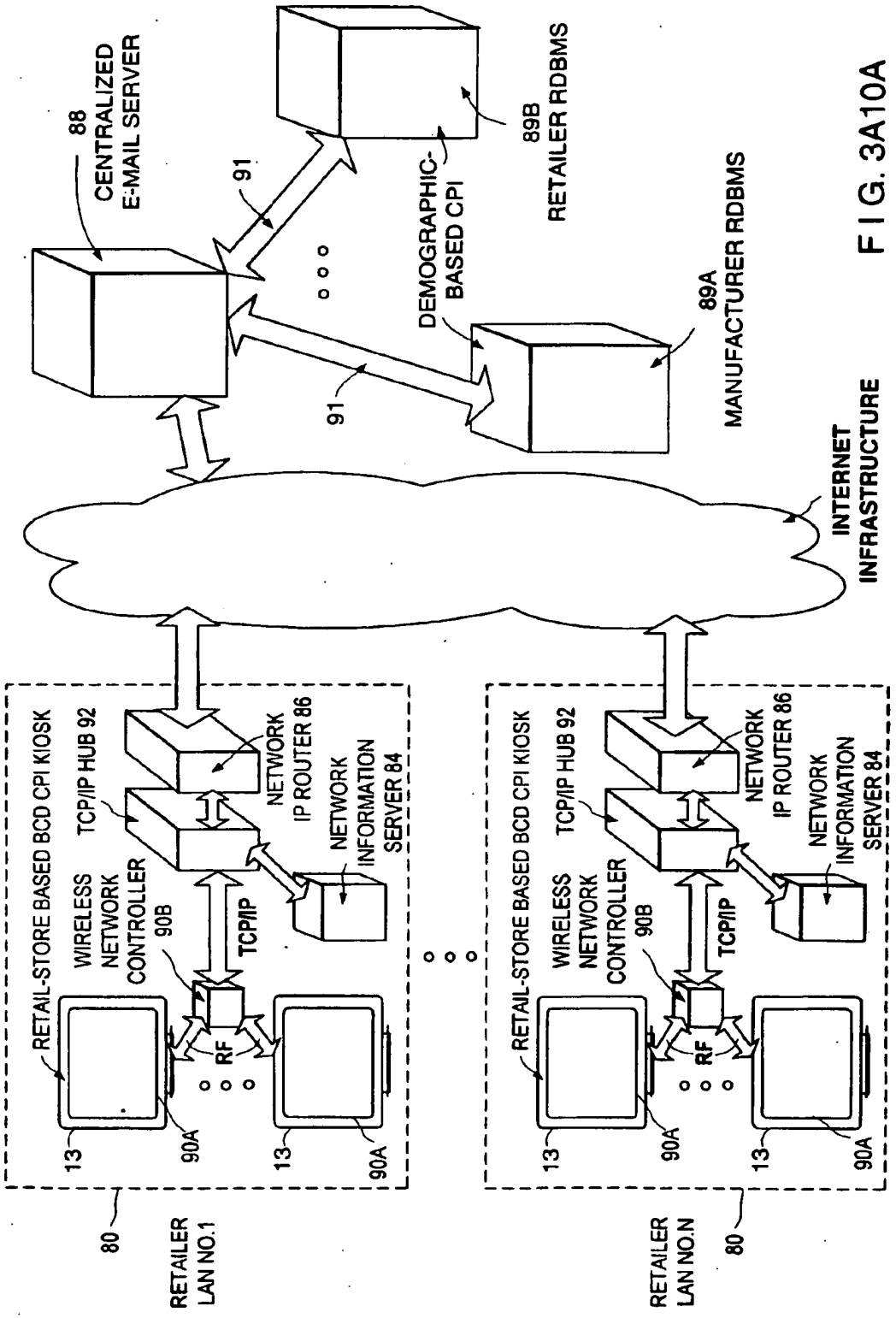

88
CENTRALIZED
E-MAIL SERVER
91
89B
RETAILER RDBMS
DEMOGRAPHIC-
BASED CPI
89A
MANUFACTURER RDBMS
INTERNET
INFRASTRUCTURE
RETAIL-STORE BASED BCD CPI KIOSK
TCP/IP HUB 92
NETWORK
IP ROUTER 86
NETWORK
INFORMATION
SERVER 84
WIRELESS
NETWORK
CONTROLLER
90B
TCP/IP
RF
13
90A
80
RETAILER
LAN NO.1
RETAILER
LAN NO.N

FIG. 3A10A

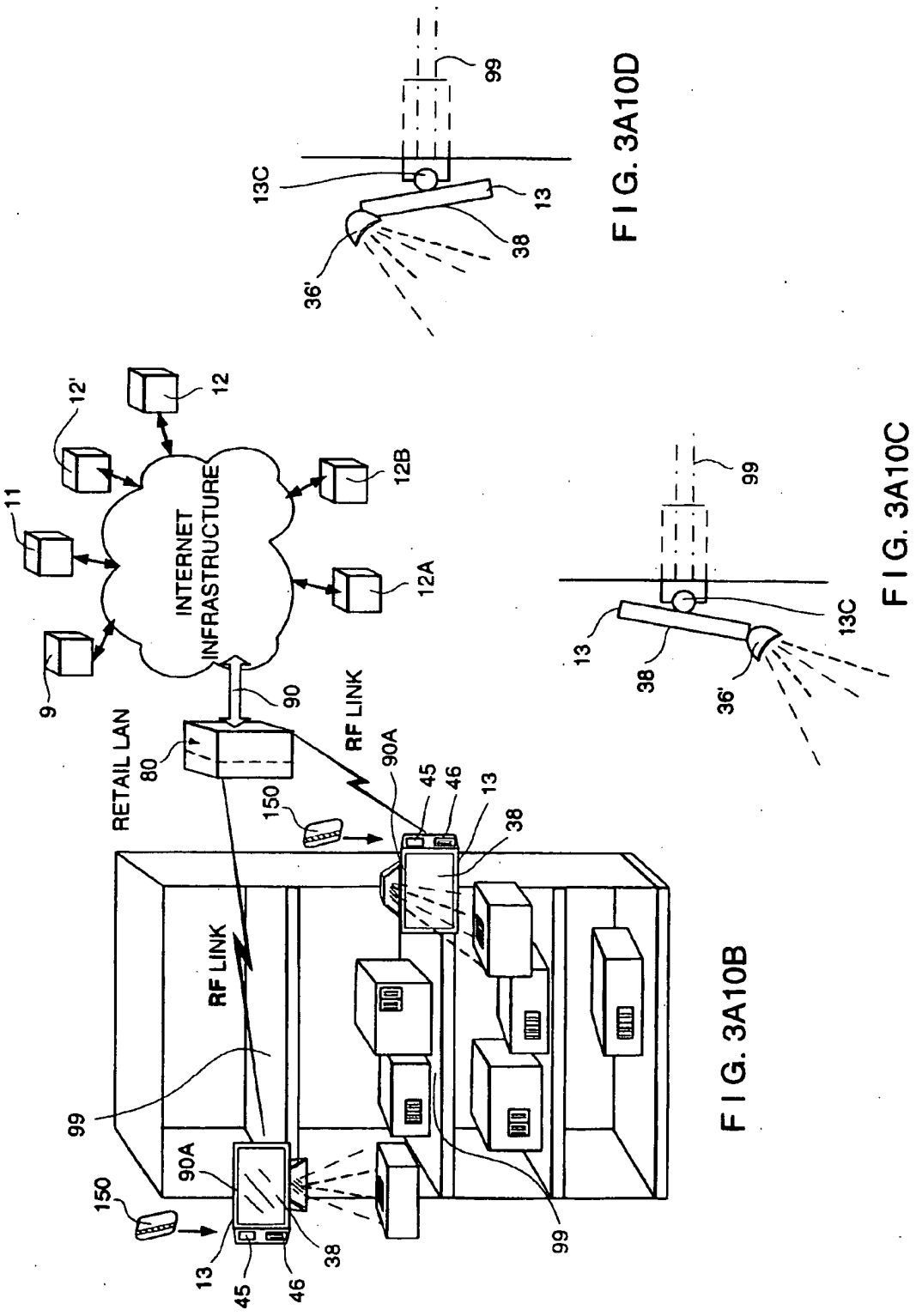


FIG. 3A10D

FIG. 3A10C

FIG. 3A10B

MASTER DATABASE TABLE

MAINTAINED BY **RETAILER** DATABASE MANAGEMENT SUBSYSTEM

RETAILER NAME	RETAILER KIOSK E-MAIL ACCNT. NO.	RETAIL KIOSK NO.	UPN	URL	DATE	DESTINATION E-MAIL (HOME)	E-MAIL DELIVERY CONFIRMATION	CONSUMER NAME	CONSUMER ADDRESS
• • •	• • •	• • •	• • •	• • •	• • •	• • •	• • •	• • •	• • •

FIG. 3A11

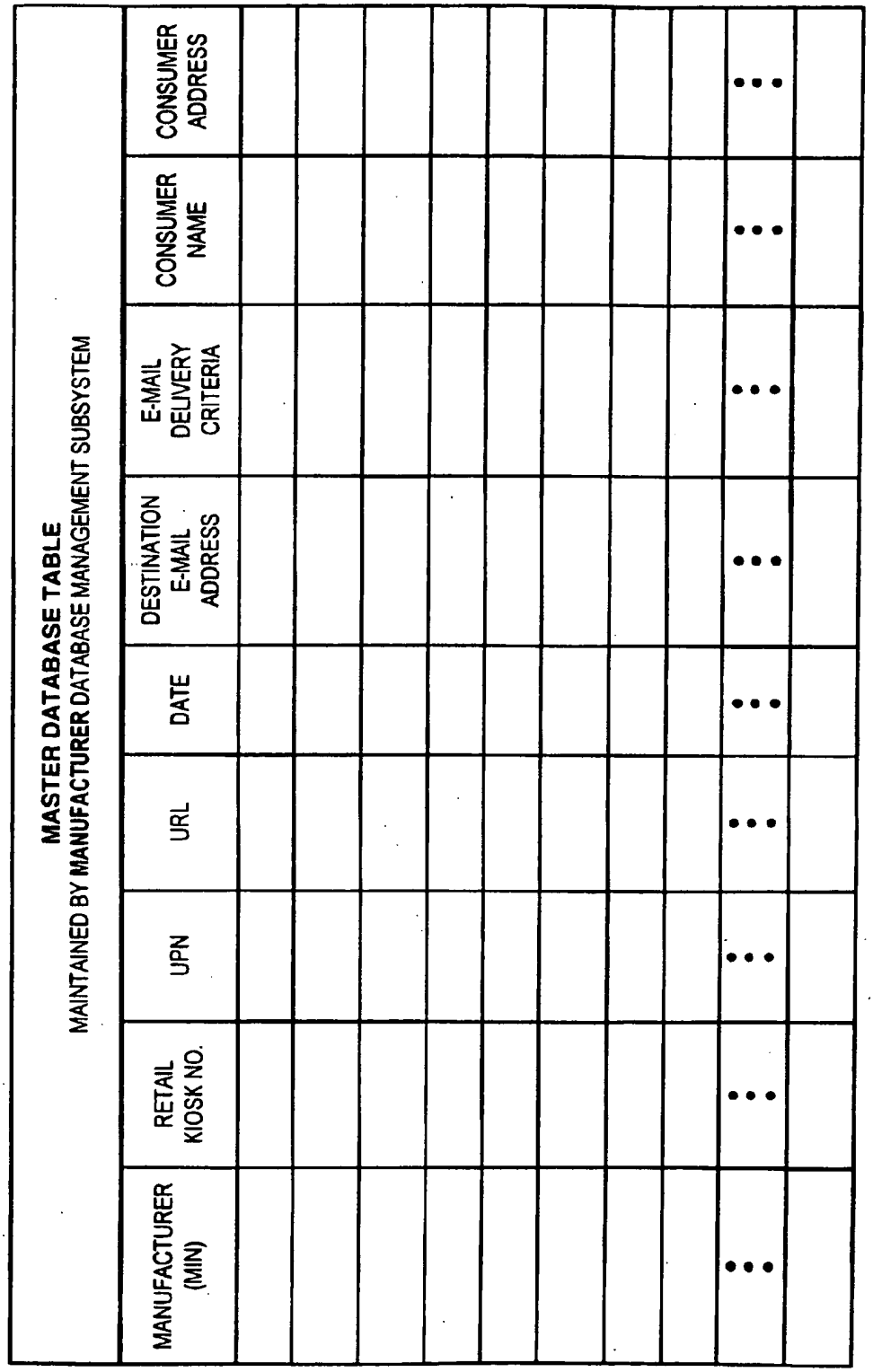

MASTER DATABASE TABLE
MAINTAINED BY MANUFACTURER DATABASE MANAGEMENT SUBSYSTEM
MANUFACTURER (MIN)
RETAIL KIOSK NO.
UPN
URL
DATE
DESTINATION E-MAIL ADDRESS
E-MAIL DELIVERY CRITERIA
CONSUMER NAME
CONSUMER ADDRESS

FIG. 3A12

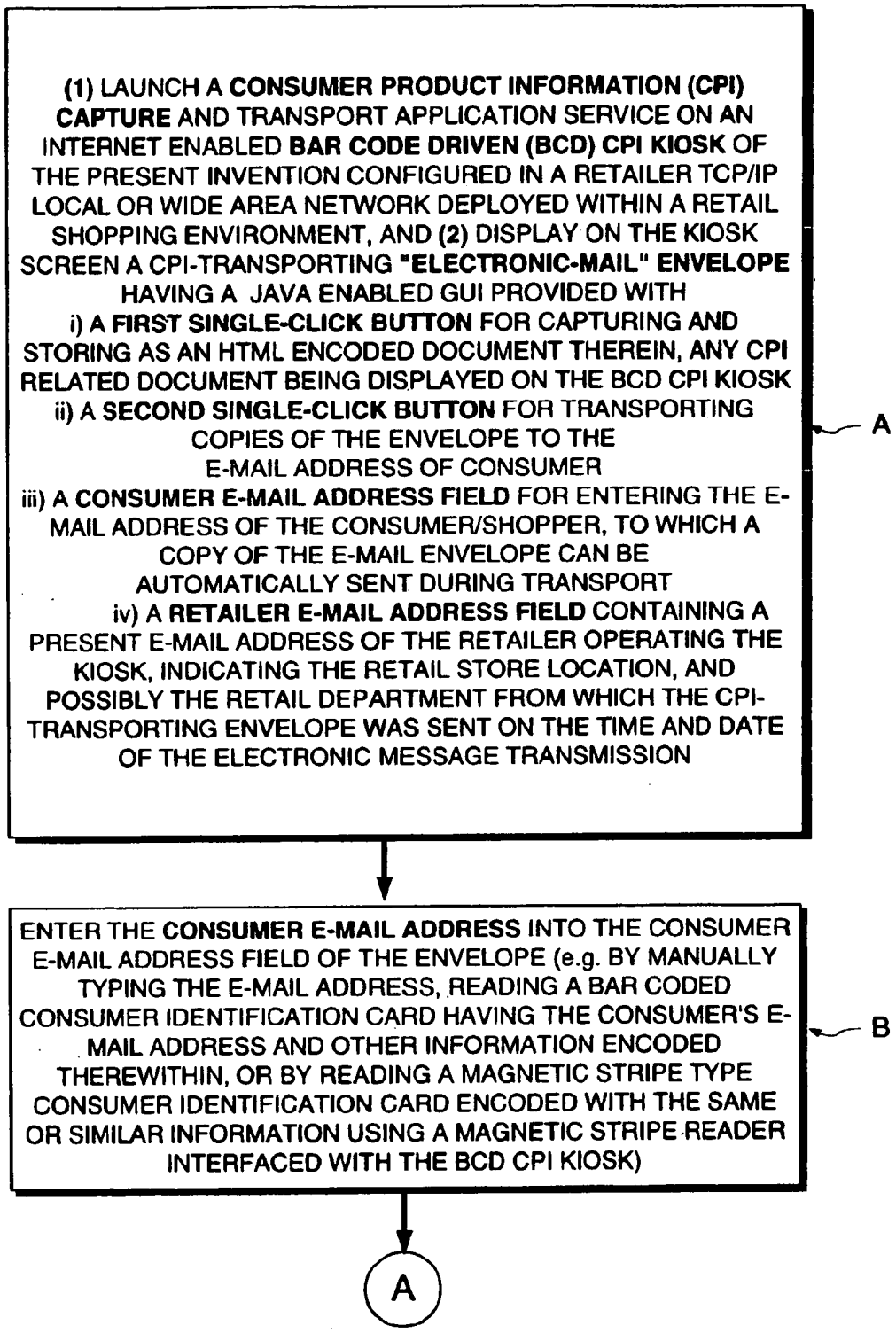

(1) LAUNCH A CONSUMER PRODUCT INFORMATION (CPI) CAPTURE AND TRANSPORT APPLICATION SERVICE ON AN INTERNET ENABLED BAR CODE DRIVEN (BCD) CPI KIOSK OF THE PRESENT INVENTION CONFIGURED IN A RETAILER TCP/IP LOCAL OR WIDE AREA NETWORK DEPLOYED WITHIN A RETAIL SHOPPING ENVIRONMENT, AND (2) DISPLAY ON THE KIOSK SCREEN A CPI-TRANSPORTING "ELECTRONIC-MAIL" ENVELOPE HAVING A JAVA ENABLED GUI PROVIDED WITH
i) A FIRST SINGLE-CLICK BUTTON FOR CAPTURING AND STORING AS AN HTML ENCODED DOCUMENT THEREIN, ANY CPI RELATED DOCUMENT BEING DISPLAYED ON THE BCD CPI KIOSK
ii) A SECOND SINGLE-CLICK BUTTON FOR TRANSPORTING COPIES OF THE ENVELOPE TO THE E-MAIL ADDRESS OF CONSUMER
iii) A CONSUMER E-MAIL ADDRESS FIELD FOR ENTERING THE E-MAIL ADDRESS OF THE CONSUMER/SHOPPER, TO WHICH A COPY OF THE E-MAIL ENVELOPE CAN BE AUTOMATICALLY SENT DURING TRANSPORT
iv) A RETAILER E-MAIL ADDRESS FIELD CONTAINING A PRESENT E-MAIL ADDRESS OF THE RETAILER OPERATING THE KIOSK, INDICATING THE RETAIL STORE LOCATION, AND POSSIBLY THE RETAIL DEPARTMENT FROM WHICH THE CPI-TRANSPORTING ENVELOPE WAS SENT ON THE TIME AND DATE OF THE ELECTRONIC MESSAGE TRANSMISSION
A
ENTER THE CONSUMER E-MAIL ADDRESS INTO THE CONSUMER E-MAIL ADDRESS FIELD OF THE ENVELOPE (e.g. BY MANUALLY TYPING THE E-MAIL ADDRESS, READING A BAR CODED CONSUMER IDENTIFICATION CARD HAVING THE CONSUMER'S E-MAIL ADDRESS AND OTHER INFORMATION ENCODED THEREWITHIN, OR BY READING A MAGNETIC STRIPE TYPE CONSUMER IDENTIFICATION CARD ENCODED WITH THE SAME OR SIMILAR INFORMATION USING A MAGNETIC STRIPE READER INTERFACED WITH THE BCD CPI KIOSK)
B
A

FIG. 3A13A

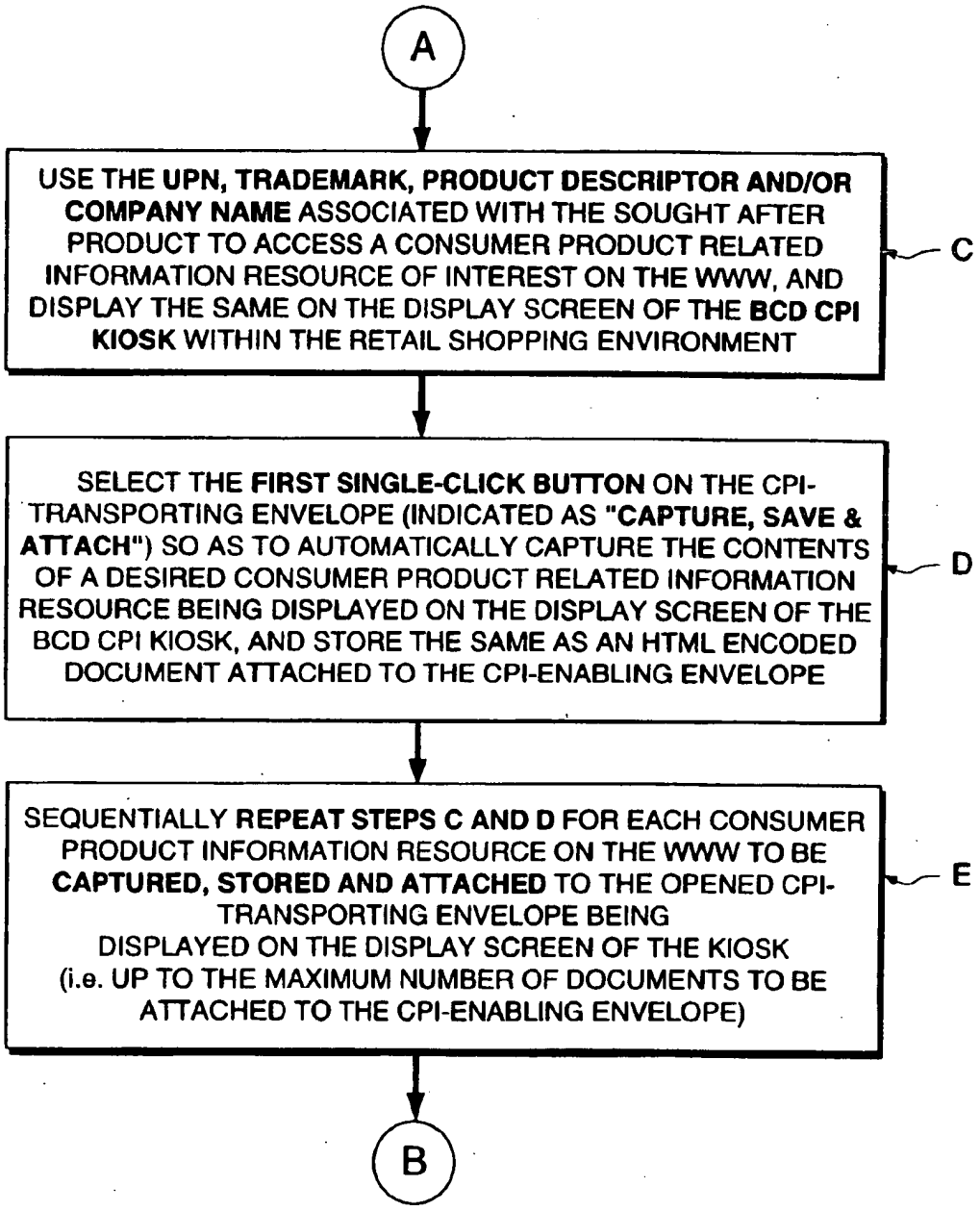

A
USE THE UPN, TRADEMARK, PRODUCT DESCRIPTOR AND/OR COMPANY NAME ASSOCIATED WITH THE SOUGHT AFTER PRODUCT TO ACCESS A CONSUMER PRODUCT RELATED INFORMATION RESOURCE OF INTEREST ON THE WWW, AND DISPLAY THE SAME ON THE DISPLAY SCREEN OF THE BCD CPI KIOSK WITHIN THE RETAIL SHOPPING ENVIRONMENT
C
SELECT THE FIRST SINGLE-CLICK BUTTON ON THE CPI-TRANSPORTING ENVELOPE (INDICATED AS "CAPTURE, SAVE & ATTACH") SO AS TO AUTOMATICALLY CAPTURE THE CONTENTS OF A DESIRED CONSUMER PRODUCT RELATED INFORMATION RESOURCE BEING DISPLAYED ON THE DISPLAY SCREEN OF THE BCD CPI KIOSK, AND STORE THE SAME AS AN HTML ENCODED DOCUMENT ATTACHED TO THE CPI-ENABLING ENVELOPE
D
SEQUENTIALLY REPEAT STEPS C AND D FOR EACH CONSUMER PRODUCT INFORMATION RESOURCE ON THE WWW TO BE CAPTURED, STORED AND ATTACHED TO THE OPENED CPI-TRANSPORTING ENVELOPE BEING DISPLAYED ON THE DISPLAY SCREEN OF THE KIOSK (i.e. UP TO THE MAXIMUM NUMBER OF DOCUMENTS TO BE ATTACHED TO THE CPI-ENABLING ENVELOPE)
E
B

FIG. 3A13B

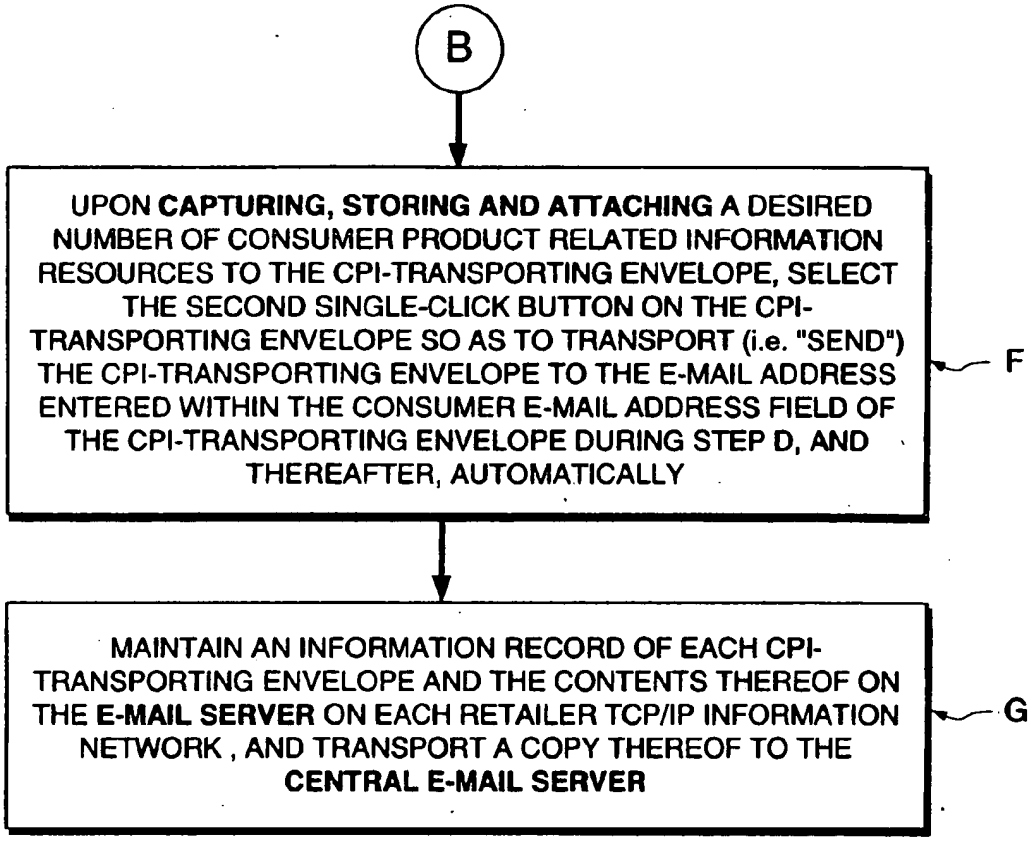

B
UPON CAPTURING, STORING AND ATTACHING A DESIRED NUMBER OF CONSUMER PRODUCT RELATED INFORMATION RESOURCES TO THE CPI-TRANSPORTING ENVELOPE, SELECT THE SECOND SINGLE-CLICK BUTTON ON THE CPI-TRANSPORTING ENVELOPE SO AS TO TRANSPORT (i.e. "SEND") THE CPI-TRANSPORTING ENVELOPE TO THE E-MAIL ADDRESS ENTERED WITHIN THE CONSUMER E-MAIL ADDRESS FIELD OF THE CPI-TRANSPORTING ENVELOPE DURING STEP D, AND THEREAFTER, AUTOMATICALLY
F
MAINTAIN AN INFORMATION RECORD OF EACH CPI-TRANSPORTING ENVELOPE AND THE CONTENTS THEREOF ON THE E-MAIL SERVER ON EACH RETAILER TCP/IP INFORMATION NETWORK , AND TRANSPORT A COPY THEREOF TO THE CENTRAL E-MAIL SERVER
G

FIG. 3A13C

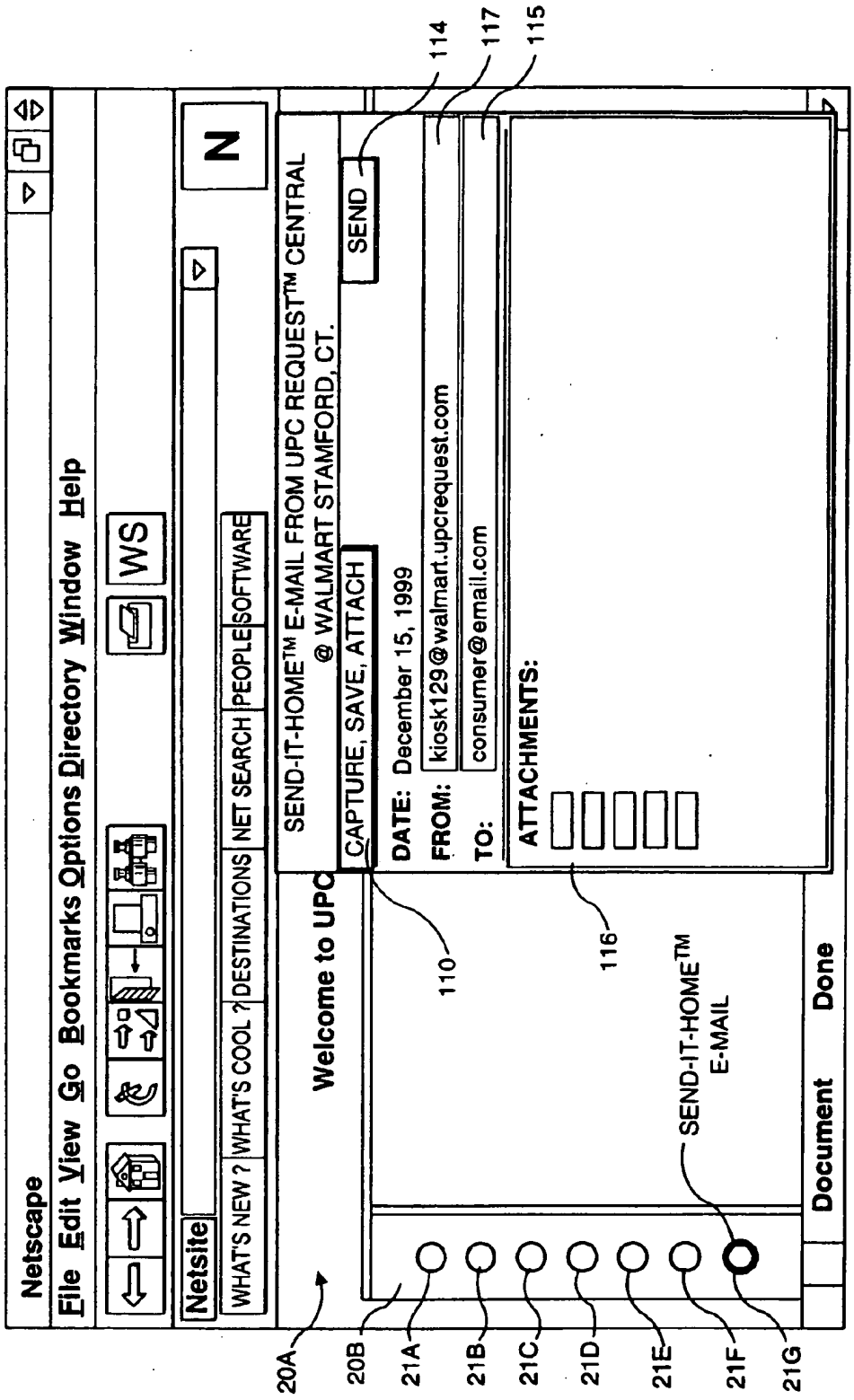

Netscape
File Edit View Go Bookmarks Options Directory Window Help
Netsite
WHAT'S NEW ? WHAT'S COOL ? DESTINATIONS NET SEARCH PEOPLE SOFTWARE
WS
N
Welcome to UPC
SEND-IT-HOME™ E-MAIL FROM UPC REQUEST™ CENTRAL
@ WALMART STAMFORD, CT.
CAPTURE, SAVE, ATTACH
SEND
DATE: December 15, 1999
FROM: kiosk129@walmart.upcrequest.com
TO: consumer@email.com
ATTACHMENTS:
SEND-IT-HOME™
E-MAIL
Document Done
114
117
115
110
116
20A
20B
21A
21B
21C
21D
21E
21F
21G

FIG. 3A14

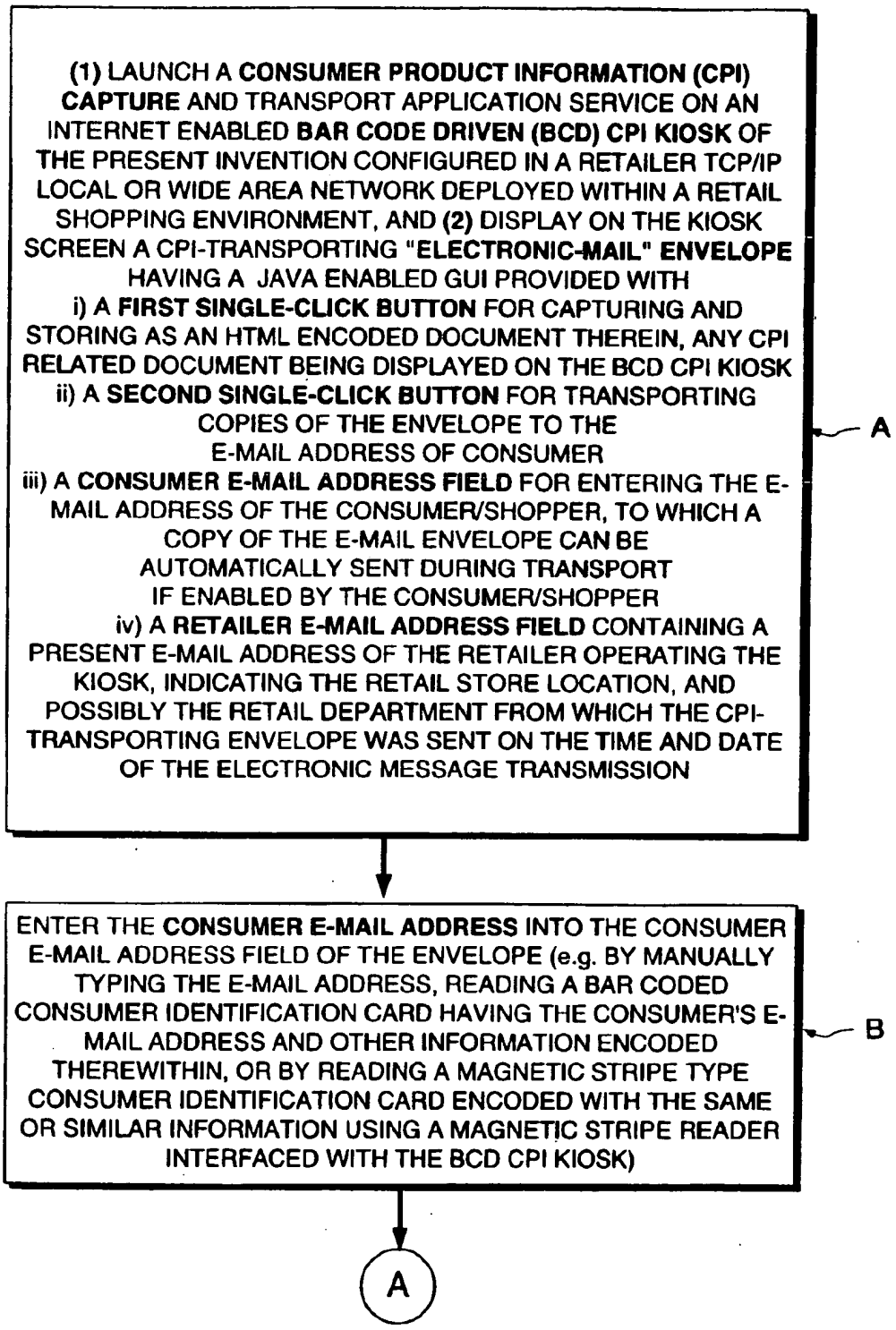

(1) LAUNCH A CONSUMER PRODUCT INFORMATION (CPI) CAPTURE AND TRANSPORT APPLICATION SERVICE ON AN INTERNET ENABLED BAR CODE DRIVEN (BCD) CPI KIOSK OF THE PRESENT INVENTION CONFIGURED IN A RETAILER TCP/IP LOCAL OR WIDE AREA NETWORK DEPLOYED WITHIN A RETAIL SHOPPING ENVIRONMENT, AND (2) DISPLAY ON THE KIOSK SCREEN A CPI-TRANSPORTING "ELECTRONIC-MAIL" ENVELOPE HAVING A JAVA ENABLED GUI PROVIDED WITH
i) A FIRST SINGLE-CLICK BUTTON FOR CAPTURING AND STORING AS AN HTML ENCODED DOCUMENT THEREIN, ANY CPI RELATED DOCUMENT BEING DISPLAYED ON THE BCD CPI KIOSK
ii) A SECOND SINGLE-CLICK BUTTON FOR TRANSPORTING COPIES OF THE ENVELOPE TO THE E-MAIL ADDRESS OF CONSUMER
iii) A CONSUMER E-MAIL ADDRESS FIELD FOR ENTERING THE E-MAIL ADDRESS OF THE CONSUMER/SHOPPER, TO WHICH A COPY OF THE E-MAIL ENVELOPE CAN BE AUTOMATICALLY SENT DURING TRANSPORT IF ENABLED BY THE CONSUMER/SHOPPER
iv) A RETAILER E-MAIL ADDRESS FIELD CONTAINING A PRESENT E-MAIL ADDRESS OF THE RETAILER OPERATING THE KIOSK, INDICATING THE RETAIL STORE LOCATION, AND POSSIBLY THE RETAIL DEPARTMENT FROM WHICH THE CPI-TRANSPORTING ENVELOPE WAS SENT ON THE TIME AND DATE OF THE ELECTRONIC MESSAGE TRANSMISSION
A
ENTER THE CONSUMER E-MAIL ADDRESS INTO THE CONSUMER E-MAIL ADDRESS FIELD OF THE ENVELOPE (e.g. BY MANUALLY TYPING THE E-MAIL ADDRESS, READING A BAR CODED CONSUMER IDENTIFICATION CARD HAVING THE CONSUMER'S E-MAIL ADDRESS AND OTHER INFORMATION ENCODED THEREWITHIN, OR BY READING A MAGNETIC STRIPE TYPE CONSUMER IDENTIFICATION CARD ENCODED WITH THE SAME OR SIMILAR INFORMATION USING A MAGNETIC STRIPE READER INTERFACED WITH THE BCD CPI KIOSK)
B
A

FIG. 3A15A

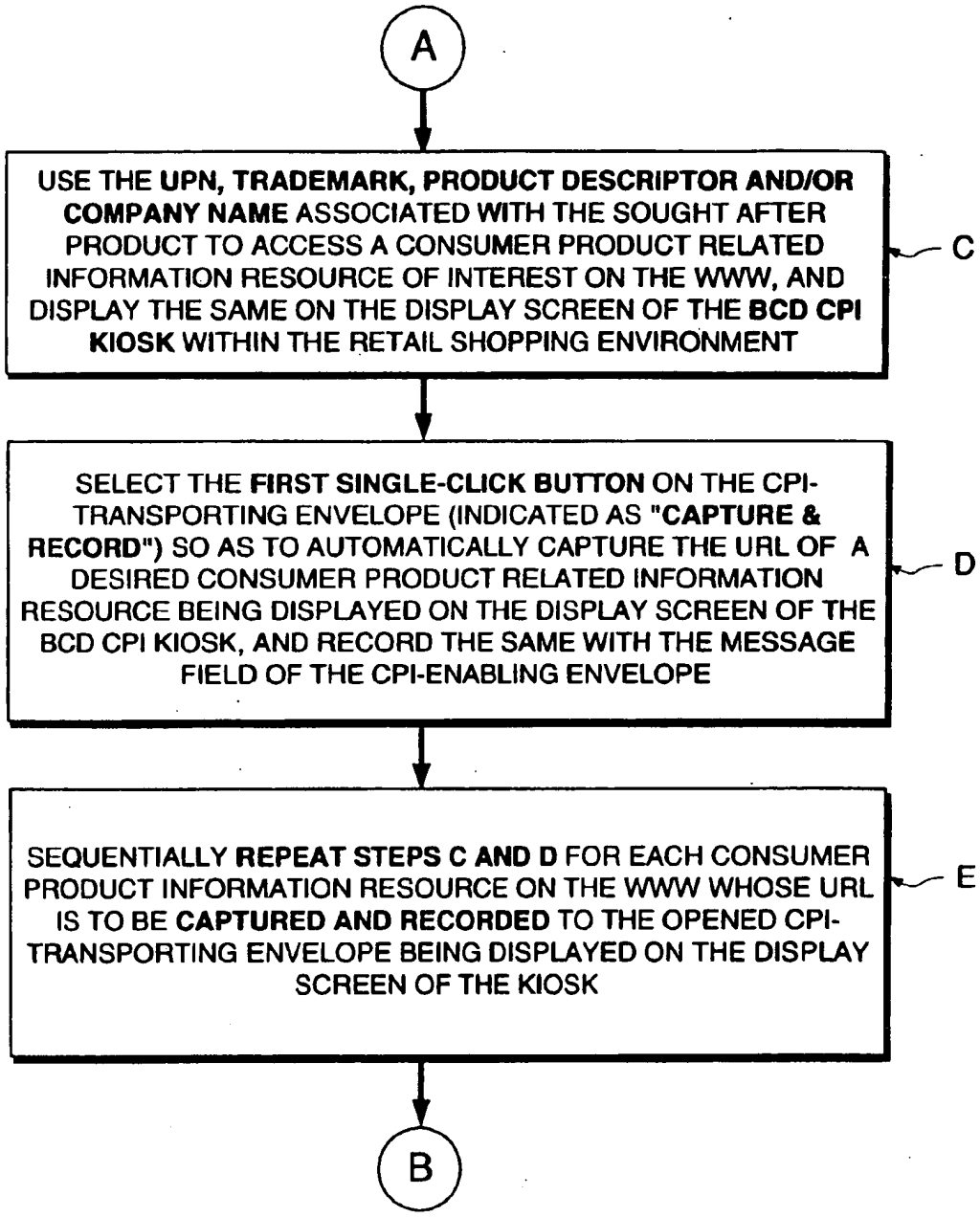

A
USE THE **UPN, TRADEMARK, PRODUCT DESCRIPTOR AND/OR COMPANY NAME** ASSOCIATED WITH THE SOUGHT AFTER PRODUCT TO ACCESS A CONSUMER PRODUCT RELATED INFORMATION RESOURCE OF INTEREST ON THE WWW, AND DISPLAY THE SAME ON THE DISPLAY SCREEN OF THE **BCD CPI KIOSK** WITHIN THE RETAIL SHOPPING ENVIRONMENT
C
SELECT THE **FIRST SINGLE-CLICK BUTTON** ON THE CPI-TRANSPORTING ENVELOPE (INDICATED AS "**CAPTURE & RECORD**") SO AS TO AUTOMATICALLY CAPTURE THE URL OF A DESIRED CONSUMER PRODUCT RELATED INFORMATION RESOURCE BEING DISPLAYED ON THE DISPLAY SCREEN OF THE BCD CPI KIOSK, AND RECORD THE SAME WITH THE MESSAGE FIELD OF THE CPI-ENABLING ENVELOPE
D
SEQUENTIALLY **REPEAT STEPS C AND D** FOR EACH CONSUMER PRODUCT INFORMATION RESOURCE ON THE WWW WHOSE URL IS TO BE **CAPTURED AND RECORDED** TO THE OPENED CPI-TRANSPORTING ENVELOPE BEING DISPLAYED ON THE DISPLAY SCREEN OF THE KIOSK
E
B

FIG. 3A15B

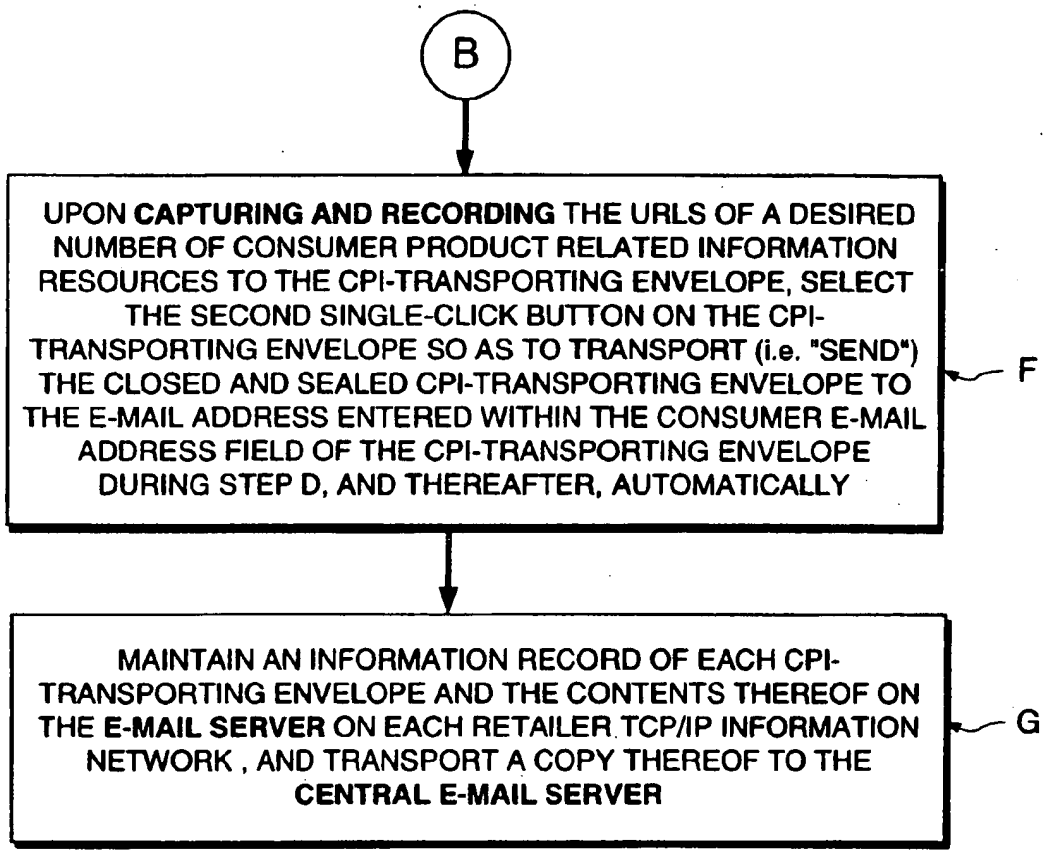

B
UPON **CAPTURING AND RECORDING** THE URLS OF A DESIRED NUMBER OF CONSUMER PRODUCT RELATED INFORMATION RESOURCES TO THE CPI-TRANSPORTING ENVELOPE, SELECT THE SECOND SINGLE-CLICK BUTTON ON THE CPI-TRANSPORTING ENVELOPE SO AS TO TRANSPORT (i.e. "SEND") THE CLOSED AND SEALED CPI-TRANSPORTING ENVELOPE TO THE E-MAIL ADDRESS ENTERED WITHIN THE CONSUMER E-MAIL ADDRESS FIELD OF THE CPI-TRANSPORTING ENVELOPE DURING STEP D, AND THEREAFTER, AUTOMATICALLY
F
MAINTAIN AN INFORMATION RECORD OF EACH CPI-TRANSPORTING ENVELOPE AND THE CONTENTS THEREOF ON THE **E-MAIL SERVER** ON EACH RETAILER TCP/IP INFORMATION NETWORK , AND TRANSPORT A COPY THEREOF TO THE **CENTRAL E-MAIL SERVER**
G

FIG. 3A15C

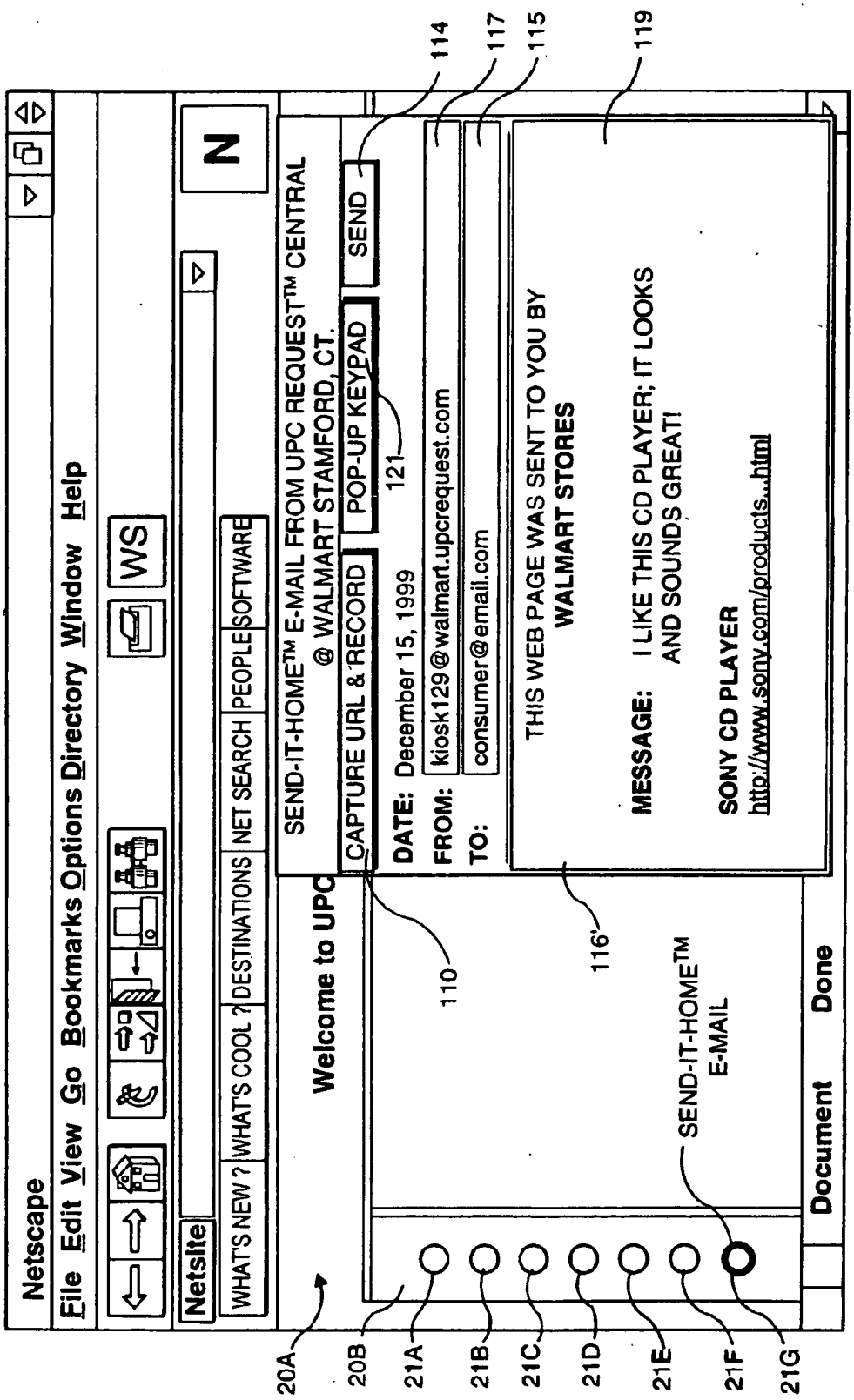

Netscape
File Edit View Go Bookmarks Options Directory Window Help
Netsite
WHAT'S NEW ? WHAT'S COOL ? DESTINATIONS NET SEARCH PEOPLE SOFTWARE
WS
N
Welcome to UPC
SEND-IT-HOME™ E-MAIL FROM UPC REQUEST™ CENTRAL
@ WALMART STAMFORD, CT.
CAPTURE URL & RECORD
POP-UP KEYPAD
SEND
DATE: December 15, 1999
FROM: kiosk129@walmart.upcrequest.com
TO: consumer@email.com
THIS WEB PAGE WAS SENT TO YOU BY
WALMART STORES
MESSAGE: I LIKE THIS CD PLAYER; IT LOOKS
AND SOUNDS GREAT!
SONY CD PLAYER
http://www.sony.com/products...html
SEND-IT-HOME™
E-MAIL
Document Done
20A
20B
21A
21B
21C
21D
21E
21F
21G
110
116'
121
114
117
115
119

FIG. 3A16

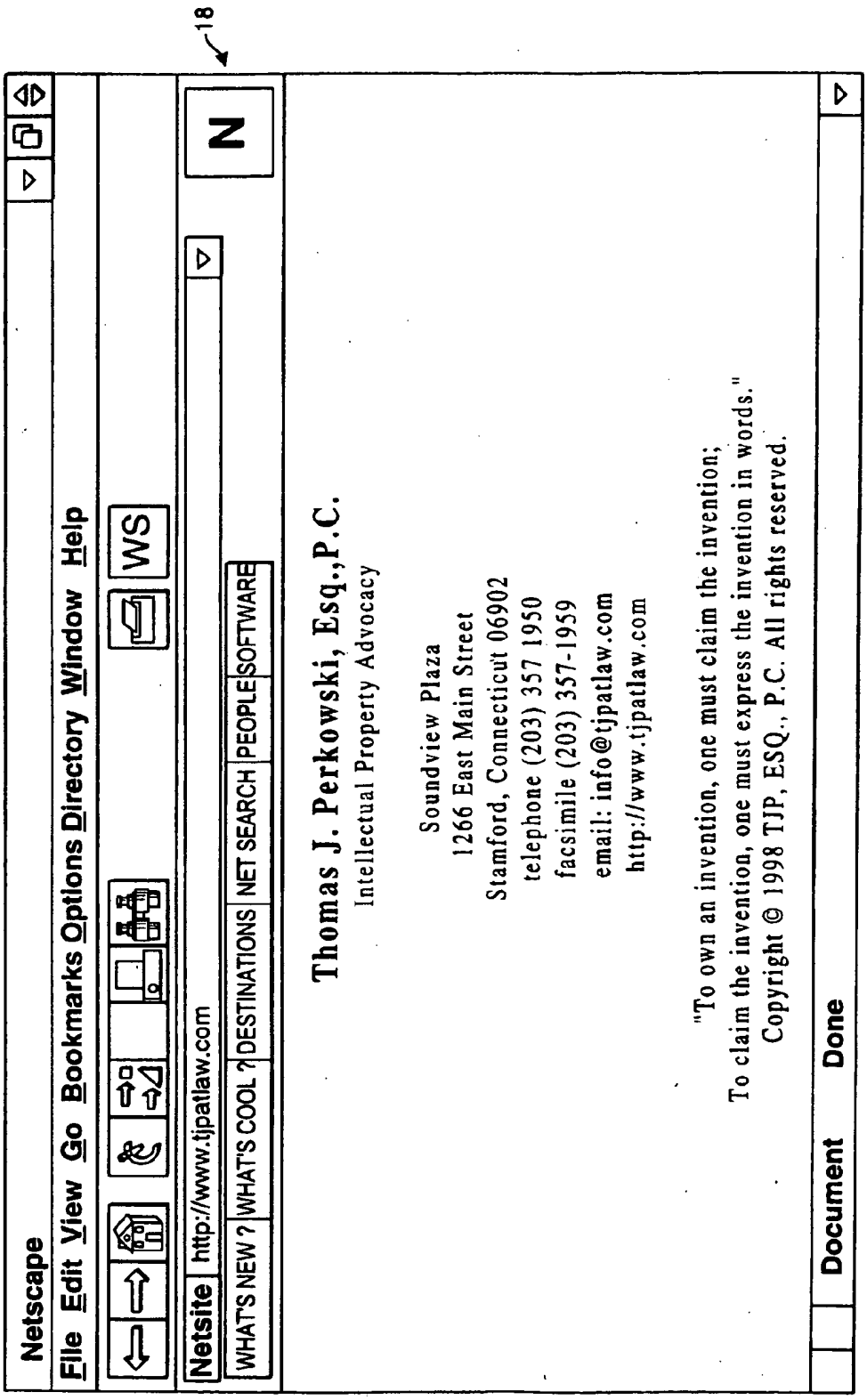

18
Netscape
File Edit View Go Bookmarks Options Directory Window Help
WS
Netsite http://www.tjpatlaw.com
WHAT'S NEW ? WHAT'S COOL ? DESTINATIONS NET SEARCH PEOPLE SOFTWARE
N
Thomas J. Perkowski, Esq., P.C.
Intellectual Property Advocacy
Soundview Plaza
1266 East Main Street
Stamford, Connecticut 06902
telephone (203) 357 1950
facsimile (203) 357-1959
email: info@tjpatlaw.com
http://www.tjpatlaw.com
"To own an invention, one must claim the invention;
To claim the invention, one must express the invention in words."
Copyright © 1998 TJP, ESQ., P.C. All rights reserved.
Document Done

FIG. 3B

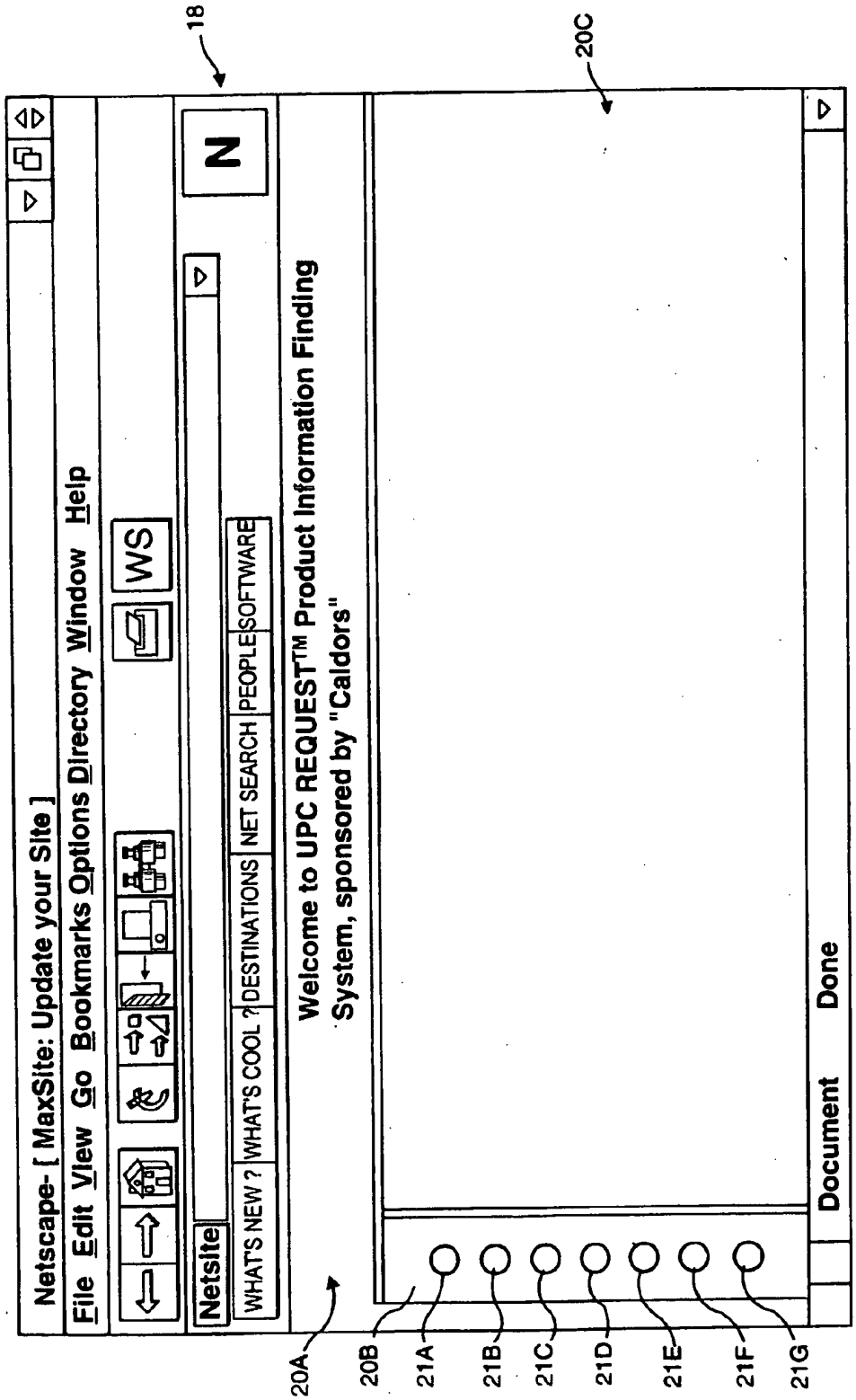
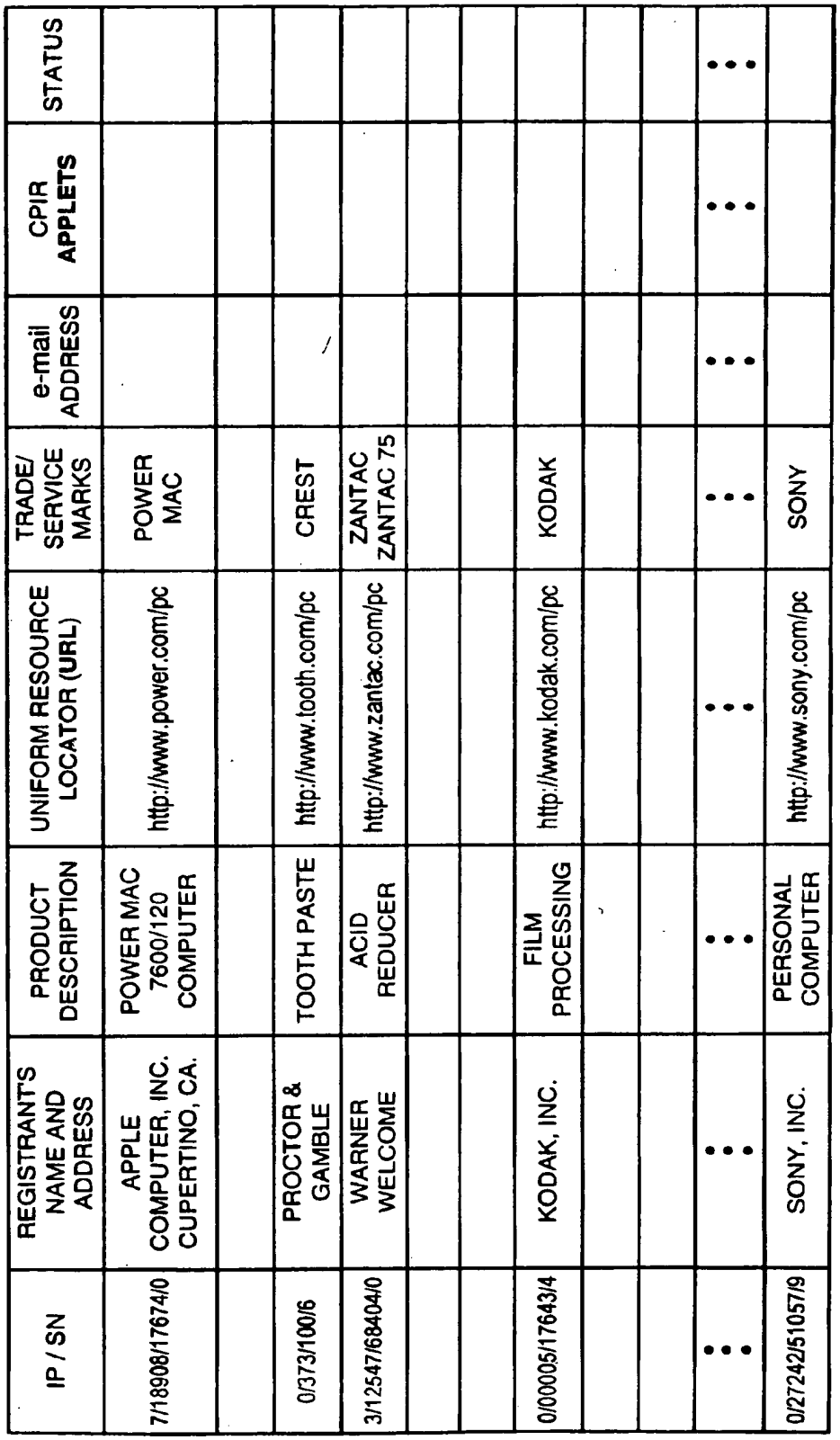

18
20C
Netscape- [MaxSite: Update your Site]
File Edit View Go Bookmarks Options Directory Window Help
Netsite
WHAT'S NEW ?
WHAT'S COOL ?
DESTINATIONS
NET SEARCH
PEOPLE
SOFTWARE
Welcome to UPC REQUEST™ Product Information Finding System, sponsored by "Caldors"
Document Done
20A
20B
21A
21B
21C
21D
21E
21F
21G

FIG. 3C



IP / SN	REGISTRANT'S NAME AND ADDRESS	PRODUCT DESCRIPTION	UNIFORM RESOURCE LOCATOR (URL)	TRADE/ SERVICE MARKS	e-mail ADDRESS	CPIR APPLETS	STATUS
7/18908/17674/0	APPLE COMPUTER, INC. CUPERTINO, CA.	POWER MAC 7600/120 COMPUTER	http://www.power.com/pc	POWER MAC			
0/373/100/6	PROCTOR & GAMBLE	TOOTH PASTE	http://www.tooth.com/pc	CREST			
3/12547/68404/0	WARNER WELCOME	ACID REDUCER	http://www.zantac.com/pc	ZANTAC ZANTAC 75			
0/00005/17643/4	KODAK, INC.	FILM PROCESSING	http://www.kodak.com/pc	KODAK			
• • •	• • •	• • •	• • •	• • •	• • •	• • •	• • •
0/27242/51057/9	SONY, INC.	PERSONAL COMPUTER	http://www.sony.com/pc	SONY			

CONSUMER PRODUCT INFO. REQUEST, CPIR-ENABLING APPLET LIBRARY

FIG. 4A1

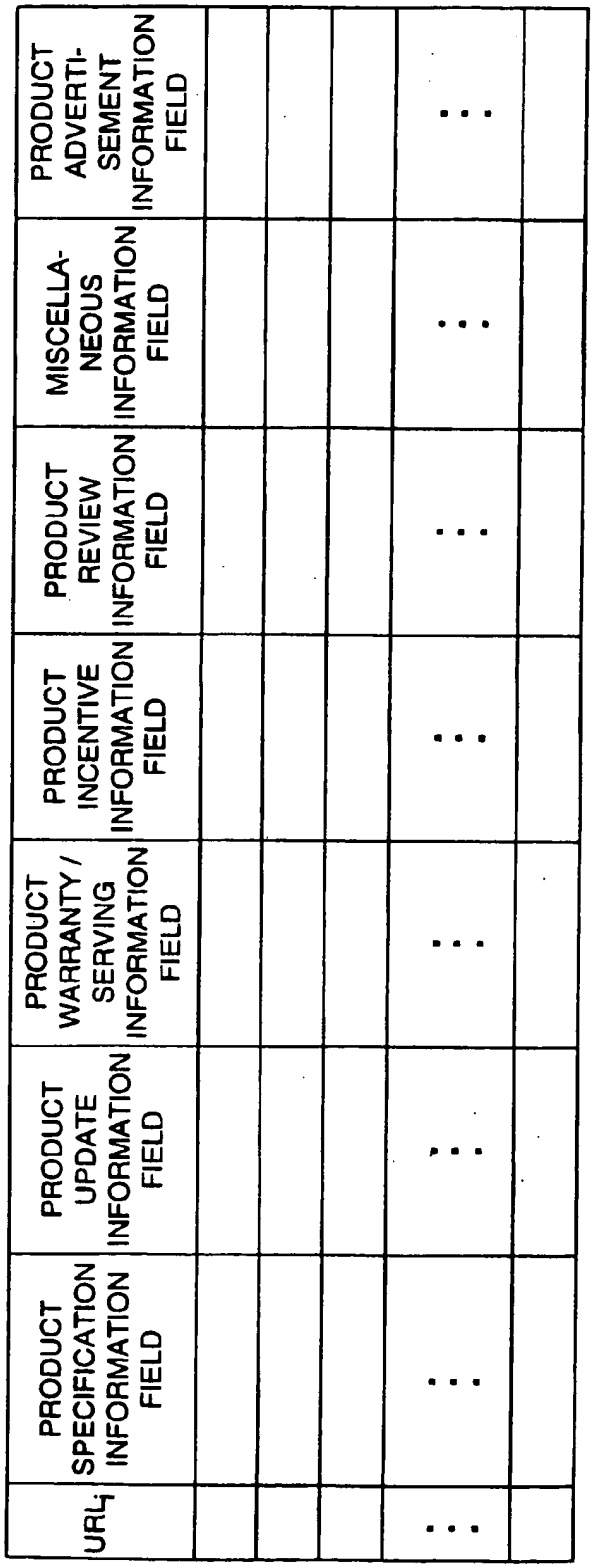
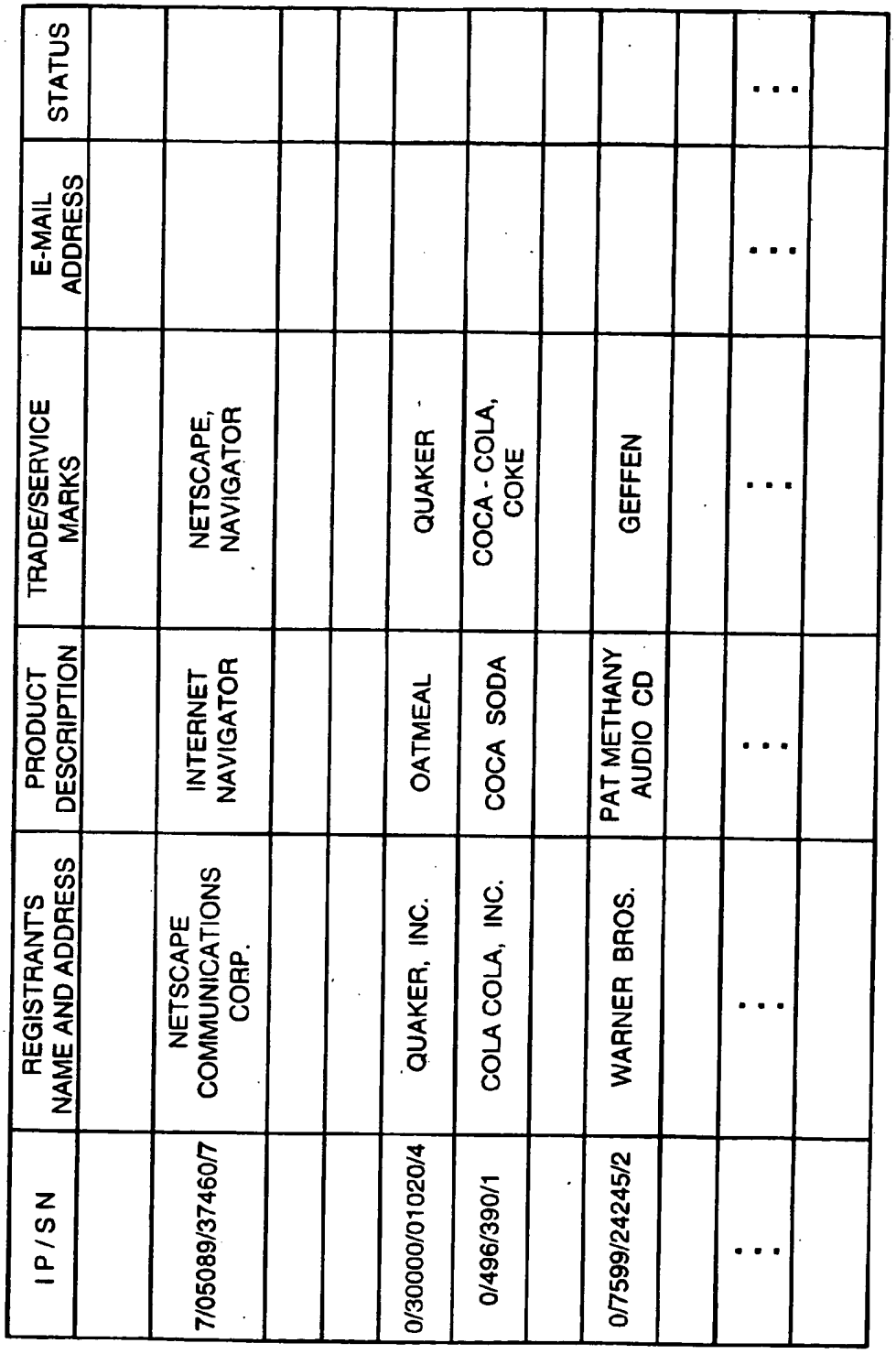

URL
PRODUCT SPECIFICATION INFORMATION FIELD
PRODUCT UPDATE INFORMATION FIELD
PRODUCT WARRANTY / SERVING INFORMATION FIELD
PRODUCT INCENTIVE INFORMATION FIELD
PRODUCT REVIEW INFORMATION FIELD
MISCELLA-NEOUS INFORMATION FIELD
PRODUCT ADVERTI-SEMENT INFORMATION FIELD

FIG. 4A2



IP/SN	REGISTRANT'S NAME AND ADDRESS	PRODUCT DESCRIPTION	TRADE/SERVICE MARKS	E-MAIL ADDRESS	STATUS
7/05089/37460/7	NETSCAPE COMMUNICATIONS CORP.	INTERNET NAVIGATOR	NETSCAPE, NAVIGATOR		
0/30000/01020/4	QUAKER, INC.	OATMEAL	QUAKER		
0/496/390/1	COLA COLA, INC.	COCA SODA	COCA - COLA, COKE		
0/7599/24245/2	WARNER BROS.	PAT METHANY AUDIO CD	GEFFEN		
. . .	. . .	. . .	. . .	. . .	. . .

FIG. 4B

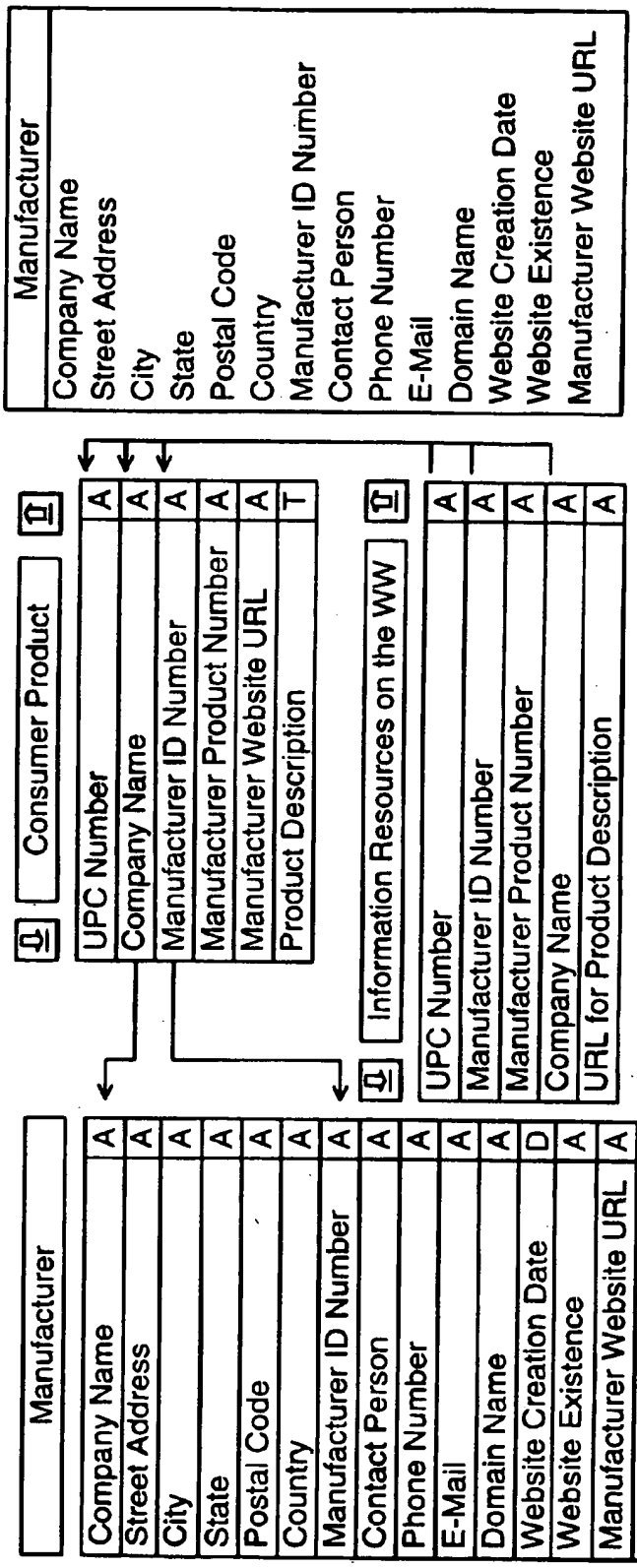


FIG. 4C1

FIG. 4C

Info. Resources on WWW
UPC Number
Manufacturer ID Number
Manufacturer Product Number
Company Name
URL For Product Description
URL For Product Manual
URL For Warranty Service
URL For WWW Advertisement
URL For WWW Advertisement
URL For WWW Advertisement
URL For Product Wholesaler
URL For Product Wholesaler
URL For Product Wholesaler
URL For Product Retailer No.1
URL For Product Retailer No.2
URL For Product Retailer No.3
URL For Direct Product Purchase
URL For Complementary Product
URL For Complementary Product
URL For Complementary Product
URL For Company Annual Report
URL For Company Stock Purchase

FIG. 4C3

Consumer Product
UPC Number
Company Name
Manufacturer ID Number
Manufacturer Product Number
Manufacturer Website URL
Product Description
Primary Trademark
Secondary Trademark
Package Type
UPC Symbol Type
Website Marking on Package
800 Consumer Phone Number

FIG. 4C2

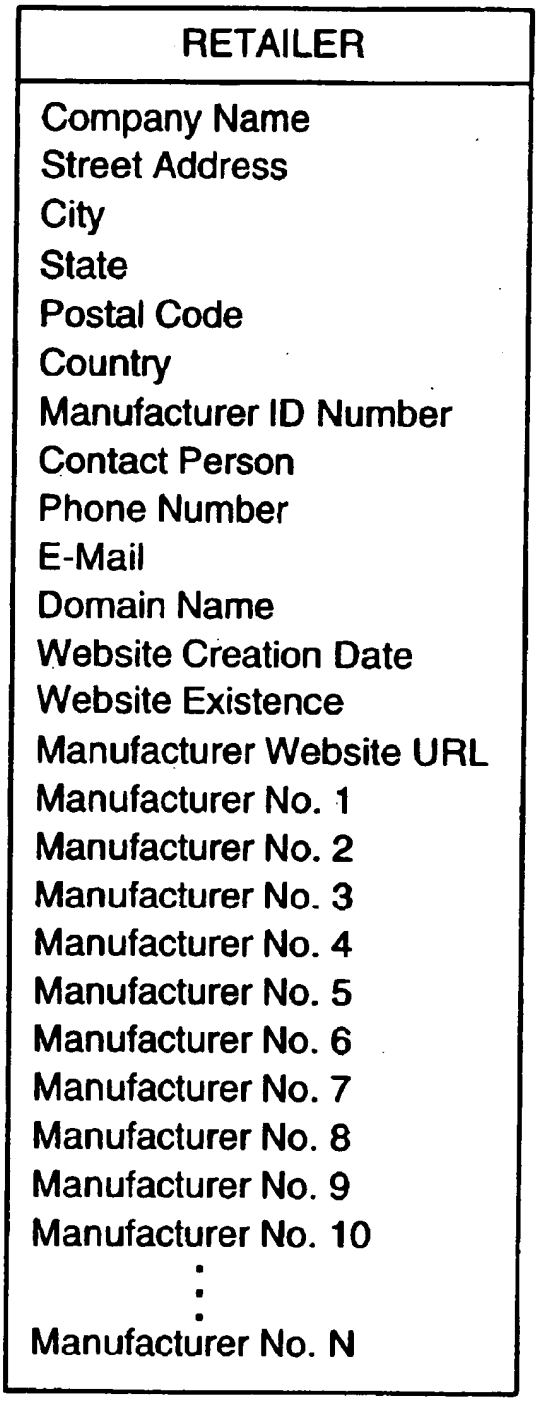

RETAILER
Company Name
Street Address
City
State
Postal Code
Country
Manufacturer ID Number
Contact Person
Phone Number
E-Mail
Domain Name
Website Creation Date
Website Existence
Manufacturer Website URL
Manufacturer No. 1
Manufacturer No. 2
Manufacturer No. 3
Manufacturer No. 4
Manufacturer No. 5
Manufacturer No. 6
Manufacturer No. 7
Manufacturer No. 8
Manufacturer No. 9
Manufacturer No. 10
Manufacturer No. N

FIG. 4D

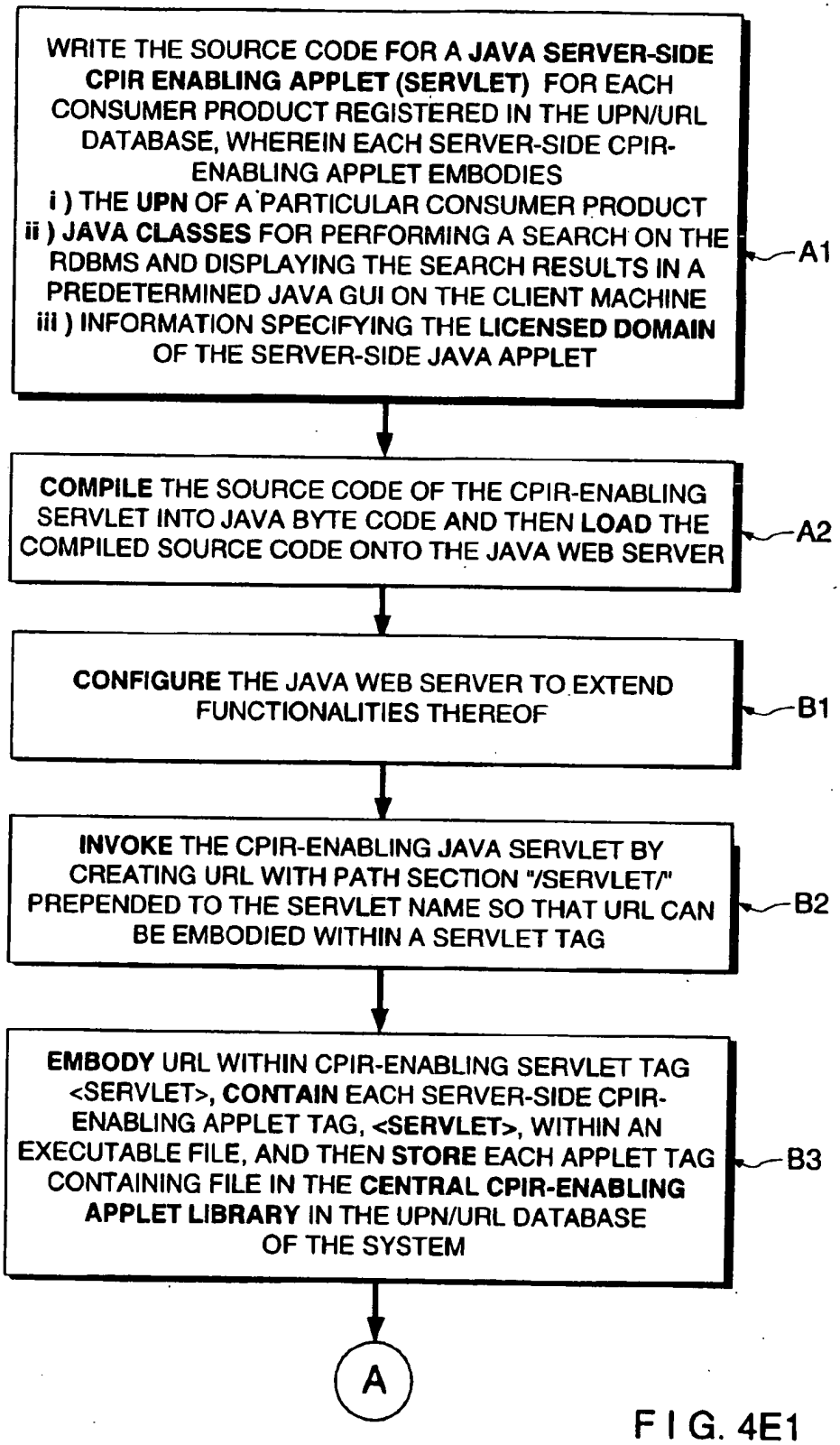

WRITE THE SOURCE CODE FOR A JAVA SERVER-SIDE CPIR ENABLING APPLET (SERVLET) FOR EACH CONSUMER PRODUCT REGISTERED IN THE UPN/URL DATABASE, WHEREIN EACH SERVER-SIDE CPIR-ENABLING APPLET EMBODIES
i) THE UPN OF A PARTICULAR CONSUMER PRODUCT
ii) JAVA CLASSES FOR PERFORMING A SEARCH ON THE RDBMS AND DISPLAYING THE SEARCH RESULTS IN A PREDETERMINED JAVA GUI ON THE CLIENT MACHINE
iii) INFORMATION SPECIFYING THE LICENSED DOMAIN OF THE SERVER-SIDE JAVA APPLET
A1
COMPILE THE SOURCE CODE OF THE CPIR-ENABLING SERVLET INTO JAVA BYTE CODE AND THEN LOAD THE COMPILED SOURCE CODE ONTO THE JAVA WEB SERVER
A2
CONFIGURE THE JAVA WEB SERVER TO EXTEND FUNCTIONALITIES THEREOF
B1
INVOKE THE CPIR-ENABLING JAVA SERVLET BY CREATING URL WITH PATH SECTION "/SERVLET/" PREPENDED TO THE SERVLET NAME SO THAT URL CAN BE EMBODIED WITHIN A SERVLET TAG
B2
EMBODY URL WITHIN CPIR-ENABLING SERVLET TAG <SERVLET>, CONTAIN EACH SERVER-SIDE CPIR-ENABLING APPLET TAG, <SERVLET>, WITHIN AN EXECUTABLE FILE, AND THEN STORE EACH APPLET TAG CONTAINING FILE IN THE CENTRAL CPIR-ENABLING APPLET LIBRARY IN THE UPN/URL DATABASE OF THE SYSTEM
B3
A

FIG. 4E1

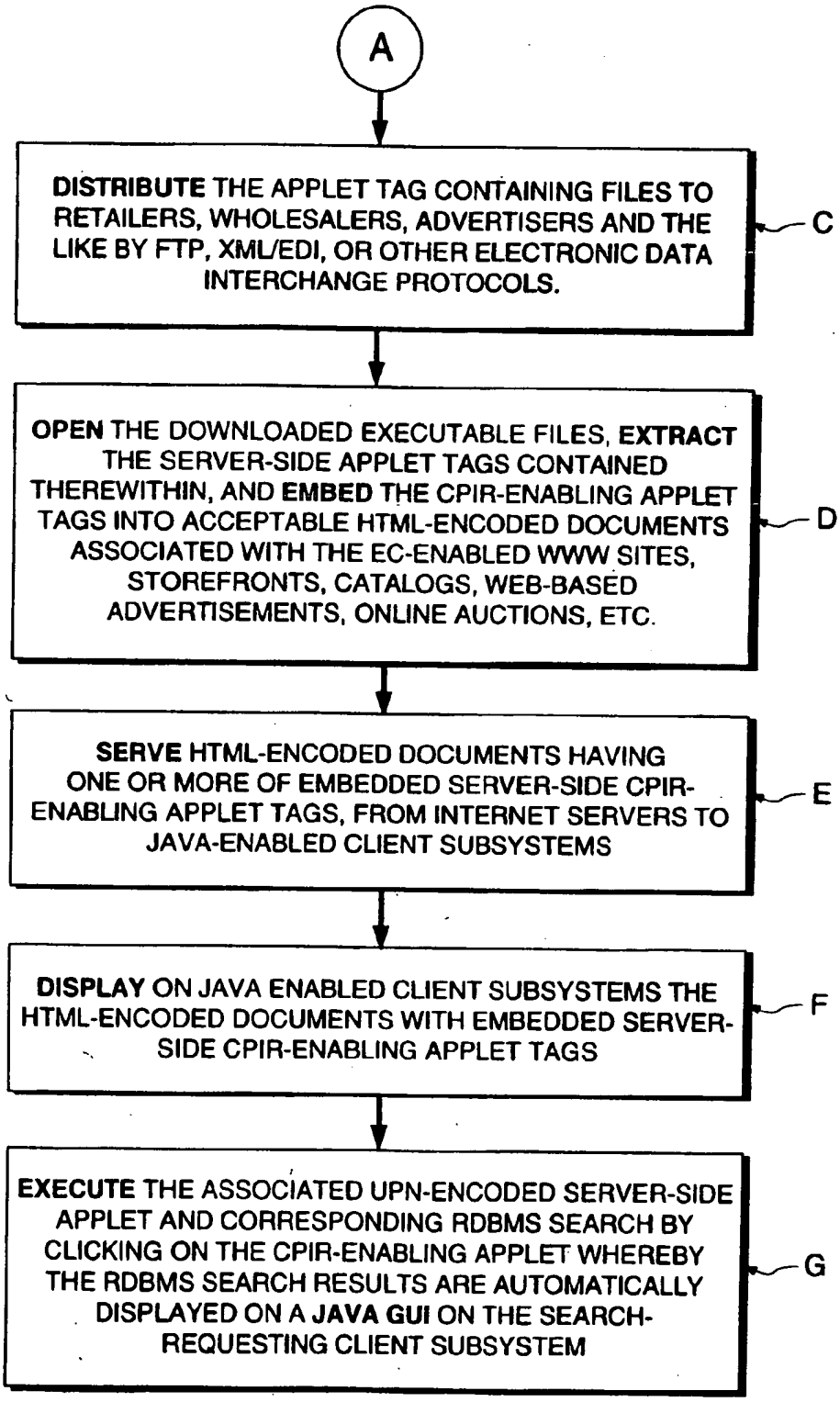

A
DISTRIBUTE THE APPLET TAG CONTAINING FILES TO RETAILERS, WHOLESALERS, ADVERTISERS AND THE LIKE BY FTP, XML/EDI, OR OTHER ELECTRONIC DATA INTERCHANGE PROTOCOLS.
C
OPEN THE DOWNLOADED EXECUTABLE FILES, EXTRACT THE SERVER-SIDE APPLET TAGS CONTAINED THEREWITHIN, AND EMBED THE CPIR-ENABLING APPLET TAGS INTO ACCEPTABLE HTML-ENCODED DOCUMENTS ASSOCIATED WITH THE EC-ENABLED WWW SITES, STOREFRONTS, CATALOGS, WEB-BASED ADVERTISEMENTS, ONLINE AUCTIONS, ETC.
D
SERVE HTML-ENCODED DOCUMENTS HAVING ONE OR MORE OF EMBEDDED SERVER-SIDE CPIR-ENABLING APPLET TAGS, FROM INTERNET SERVERS TO JAVA-ENABLED CLIENT SUBSYSTEMS
E
DISPLAY ON JAVA ENABLED CLIENT SUBSYSTEMS THE HTML-ENCODED DOCUMENTS WITH EMBEDDED SERVER-SIDE CPIR-ENABLING APPLET TAGS
F
EXECUTE THE ASSOCIATED UPN-ENCODED SERVER-SIDE APPLET AND CORRESPONDING RDBMS SEARCH BY CLICKING ON THE CPIR-ENABLING APPLET WHEREBY THE RDBMS SEARCH RESULTS ARE AUTOMATICALLY DISPLAYED ON A JAVA GUI ON THE SEARCH-REQUESTING CLIENT SUBSYSTEM
G

FIG. 4E2

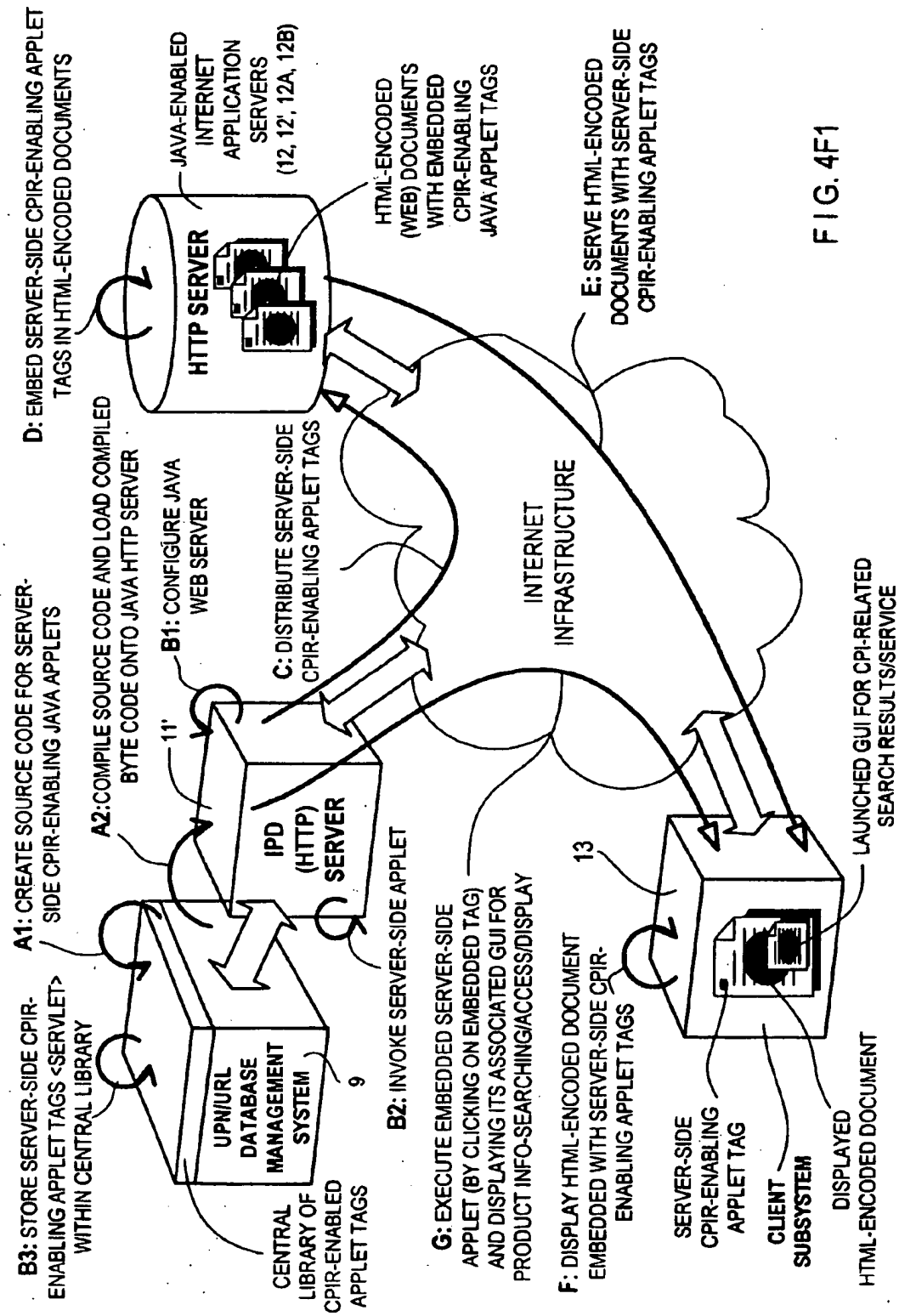

A1: CREATE SOURCE CODE FOR SERVER-SIDE CPIR-ENABLING JAVA APPLETS
A2: COMPILE SOURCE CODE AND LOAD COMPILED BYTE CODE ONTO JAVA HTTP SERVER
B1: CONFIGURE JAVA WEB SERVER
B2: INVOKE SERVER-SIDE APPLET
B3: STORE SERVER-SIDE CPIR-ENABLING APPLET TAGS <SERVLET> WITHIN CENTRAL LIBRARY
C: DISTRIBUTE SERVER-SIDE CPIR-ENABLING APPLET TAGS
D: EMBED SERVER-SIDE CPIR-ENABLING APPLET TAGS IN HTML-ENCODED DOCUMENTS
E: SERVE HTML-ENCODED DOCUMENTS WITH SERVER-SIDE CPIR-ENABLING APPLET TAGS
F: DISPLAY HTML-ENCODED DOCUMENT EMBEDDED WITH SERVER-SIDE CPIR-ENABLING APPLET TAGS
G: EXECUTE EMBEDDED SERVER-SIDE APPLET (BY CLICKING ON EMBEDDED TAG) AND DISPLAYING ITS ASSOCIATED GUI FOR PRODUCT INFO-SEARCHING/ACCESS/DISPLAY
HTTP SERVER
JAVA-ENABLED INTERNET APPLICATION SERVERS (12, 12', 12A, 12B)
HTML-ENCODED (WEB) DOCUMENTS WITH EMBEDDED CPIR-ENABLING JAVA APPLET TAGS
INTERNET INFRASTRUCTURE
IPD (HTTP) SERVER
11'
UPN/URL DATABASE MANAGEMENT SYSTEM
9
CENTRAL LIBRARY OF CPIR-ENABLED APPLET TAGS
13
SERVER-SIDE CPIR-ENABLING APPLET TAG
CLIENT SUBSYSTEM
DISPLAYED HTML-ENCODED DOCUMENT
LAUNCHED GUI FOR CPI-RELATED SEARCH RESULTS/SERVICE

FIG. 4F1

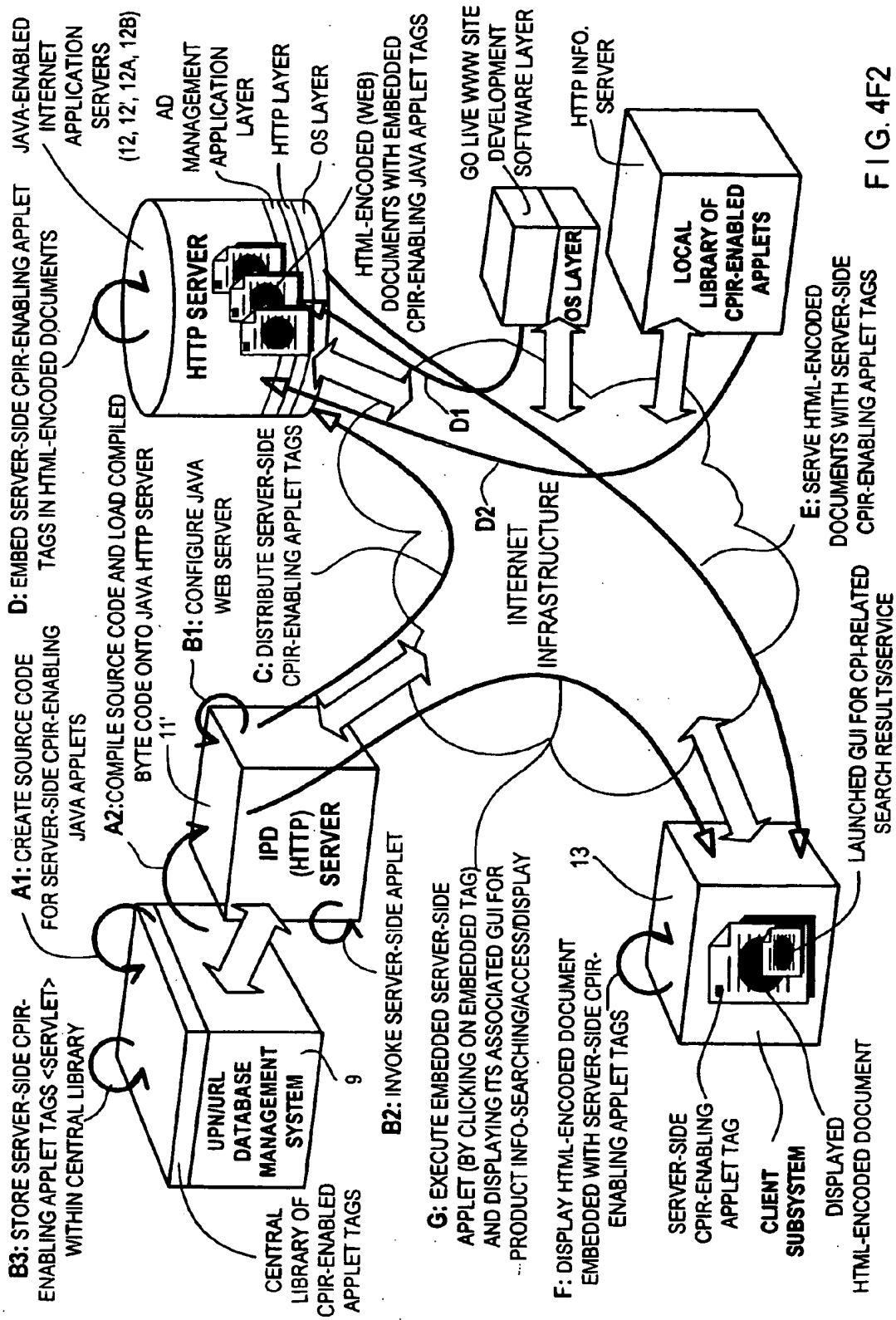

HTTP SERVER
AD MANAGEMENT APPLICATION LAYER
HTTP LAYER
OS LAYER
JAVA-ENABLED INTERNET APPLICATION SERVERS (12, 12', 12A, 12B)
D: EMBED SERVER-SIDE CPIR-ENABLING APPLET TAGS IN HTML-ENCODED DOCUMENTS
HTML-ENCODED (WEB) DOCUMENTS WITH EMBEDDED CPIR-ENABLING JAVA APPLET TAGS
GO LIVE WWW SITE DEVELOPMENT SOFTWARE LAYER
OS LAYER
HTTP INFO. SERVER
LOCAL LIBRARY OF CPIR-ENABLED APPLETS
D1
D2
INTERNET INFRASTRUCTURE
E: SERVE HTML-ENCODED DOCUMENTS WITH SERVER-SIDE CPIR-ENABLING APPLET TAGS
A1: CREATE SOURCE CODE FOR SERVER-SIDE CPIR-ENABLING JAVA APPLETS
A2: COMPILE SOURCE CODE AND LOAD COMPILED BYTE CODE ONTO JAVA HTTP SERVER
B1: CONFIGURE JAVA WEB SERVER
C: DISTRIBUTE SERVER-SIDE CPIR-ENABLING APPLET TAGS
11'
IPD (HTTP) SERVER
B2: INVOKE SERVER-SIDE APPLET
G: EXECUTE EMBEDDED SERVER-SIDE APPLET (BY CLICKING ON EMBEDDED TAG) AND DISPLAYING ITS ASSOCIATED GUI FOR PRODUCT INFO-SEARCHING/ACCESS/DISPLAY
13
F: DISPLAY HTML-ENCODED DOCUMENT EMBEDDED WITH SERVER-SIDE CPIR-ENABLING APPLET TAGS
LAUNCHED GUI FOR CPI-RELATED SEARCH RESULTS/SERVICE
B3: STORE SERVER-SIDE CPIR-ENABLING APPLET TAGS <SERVLET> WITHIN CENTRAL LIBRARY
UPN/URL DATABASE MANAGEMENT SYSTEM
9
CENTRAL LIBRARY OF CPIR-ENABLED APPLET TAGS
SERVER-SIDE CPIR-ENABLING APPLET TAG
CLIENT SUBSYSTEM
DISPLAYED HTML-ENCODED DOCUMENT

FIG. 4F2

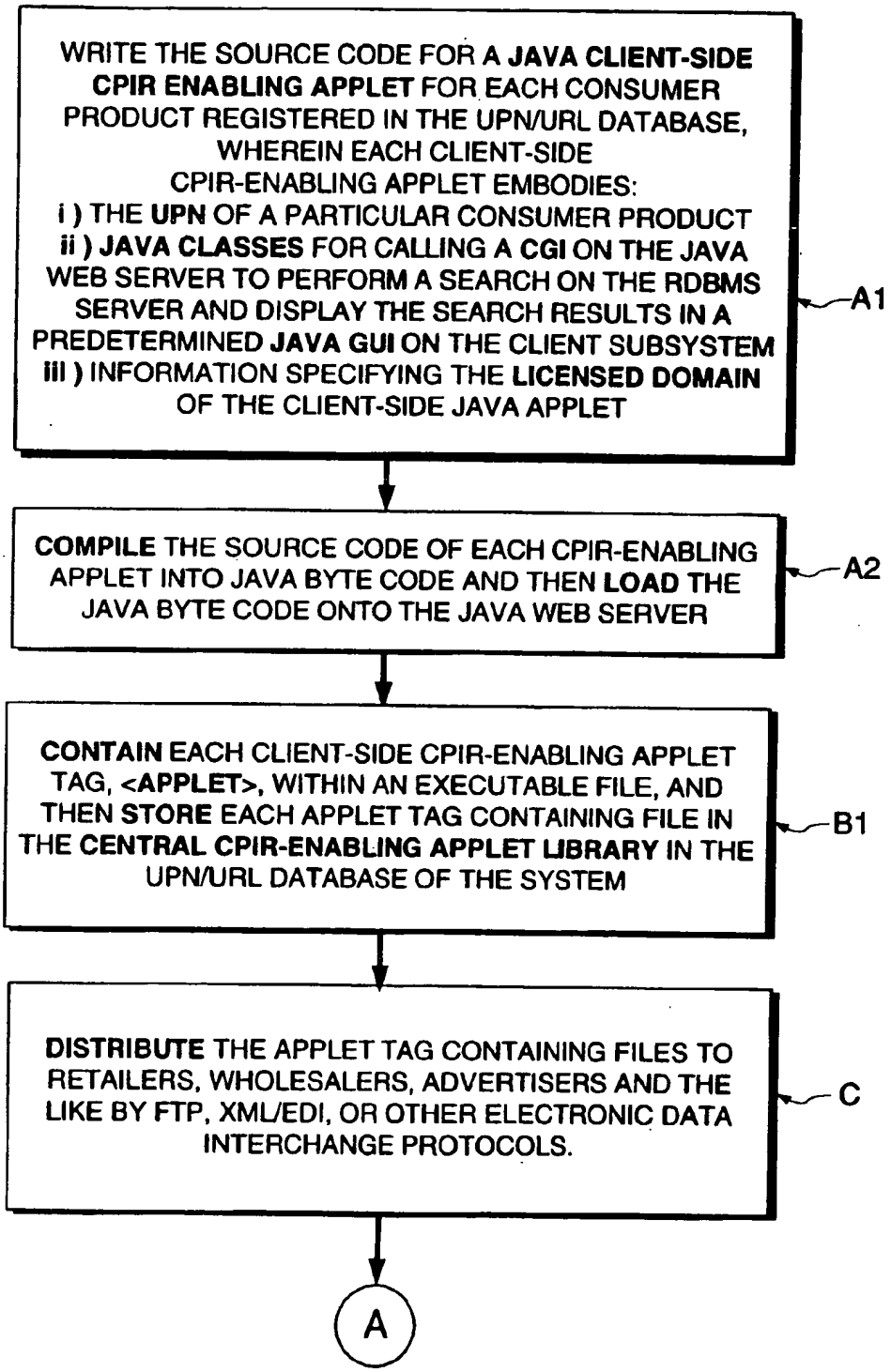

WRITE THE SOURCE CODE FOR A JAVA CLIENT-SIDE CPIR ENABLING APPLET FOR EACH CONSUMER PRODUCT REGISTERED IN THE UPN/URL DATABASE, WHEREIN EACH CLIENT-SIDE CPIR-ENABLING APPLET EMBODIES:
i) THE UPN OF A PARTICULAR CONSUMER PRODUCT
ii) JAVA CLASSES FOR CALLING A CGI ON THE JAVA WEB SERVER TO PERFORM A SEARCH ON THE RDBMS SERVER AND DISPLAY THE SEARCH RESULTS IN A PREDETERMINED JAVA GUI ON THE CLIENT SUBSYSTEM
iii) INFORMATION SPECIFYING THE LICENSED DOMAIN OF THE CLIENT-SIDE JAVA APPLET
A1
COMPILE THE SOURCE CODE OF EACH CPIR-ENABLING APPLET INTO JAVA BYTE CODE AND THEN LOAD THE JAVA BYTE CODE ONTO THE JAVA WEB SERVER
A2
CONTAIN EACH CLIENT-SIDE CPIR-ENABLING APPLET TAG, <APPLET>, WITHIN AN EXECUTABLE FILE, AND THEN STORE EACH APPLET TAG CONTAINING FILE IN THE CENTRAL CPIR-ENABLING APPLET LIBRARY IN THE UPN/URL DATABASE OF THE SYSTEM
B1
DISTRIBUTE THE APPLET TAG CONTAINING FILES TO RETAILERS, WHOLESALERS, ADVERTISERS AND THE LIKE BY FTP, XML/EDI, OR OTHER ELECTRONIC DATA INTERCHANGE PROTOCOLS.
C
A

FIG. 4G1

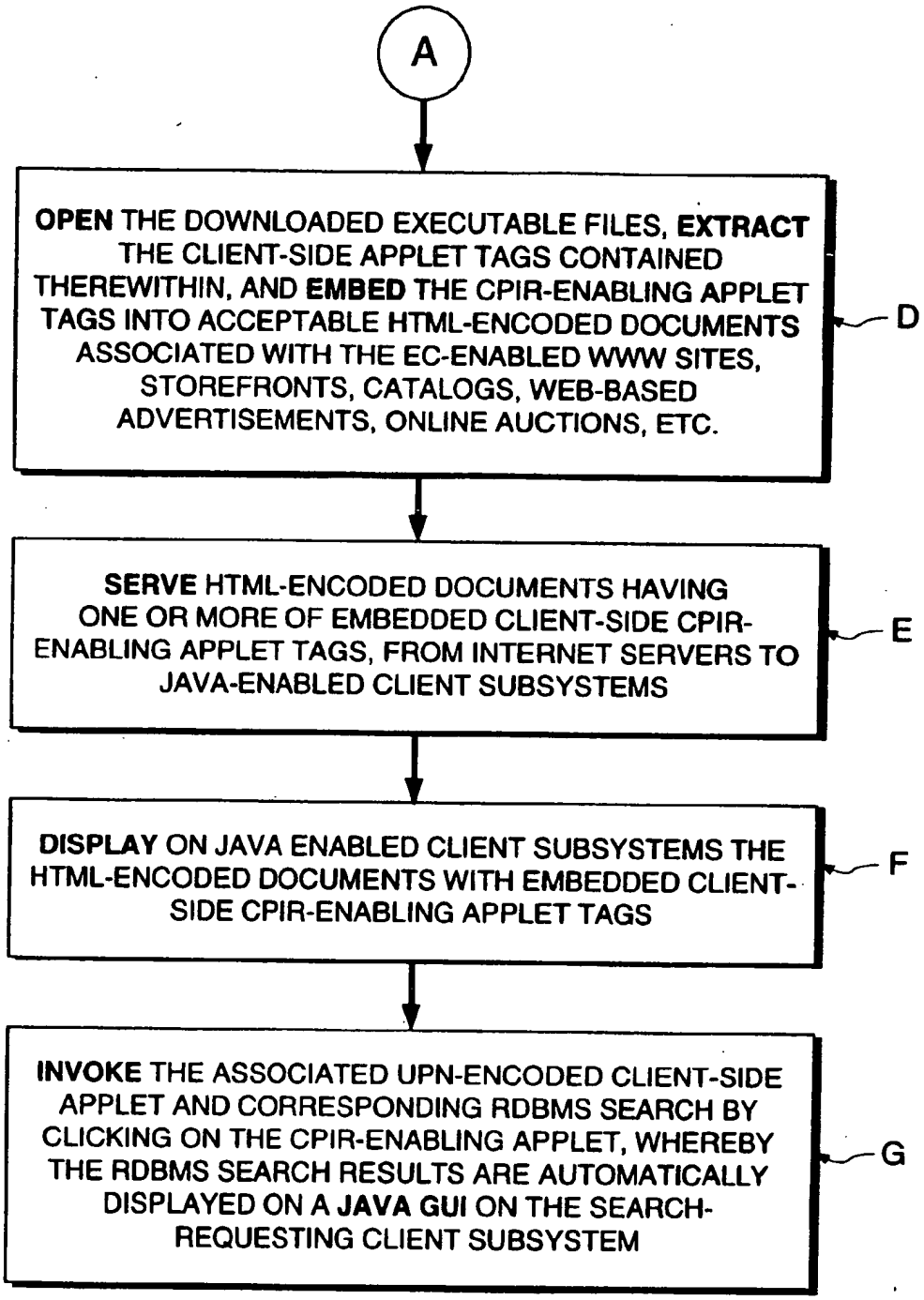

A
OPEN THE DOWNLOADED EXECUTABLE FILES, EXTRACT THE CLIENT-SIDE APPLET TAGS CONTAINED THEREWITHIN, AND EMBED THE CPIR-ENABLING APPLET TAGS INTO ACCEPTABLE HTML-ENCODED DOCUMENTS ASSOCIATED WITH THE EC-ENABLED WWW SITES, STOREFRONTS, CATALOGS, WEB-BASED ADVERTISEMENTS, ONLINE AUCTIONS, ETC.
D
SERVE HTML-ENCODED DOCUMENTS HAVING ONE OR MORE OF EMBEDDED CLIENT-SIDE CPIR-ENABLING APPLET TAGS, FROM INTERNET SERVERS TO JAVA-ENABLED CLIENT SUBSYSTEMS
E
DISPLAY ON JAVA ENABLED CLIENT SUBSYSTEMS THE HTML-ENCODED DOCUMENTS WITH EMBEDDED CLIENT-SIDE CPIR-ENABLING APPLET TAGS
F
INVOKE THE ASSOCIATED UPN-ENCODED CLIENT-SIDE APPLET AND CORRESPONDING RDBMS SEARCH BY CLICKING ON THE CPIR-ENABLING APPLET, WHEREBY THE RDBMS SEARCH RESULTS ARE AUTOMATICALLY DISPLAYED ON A JAVA GUI ON THE SEARCH-REQUESTING CLIENT SUBSYSTEM
G

FIG. 4G2

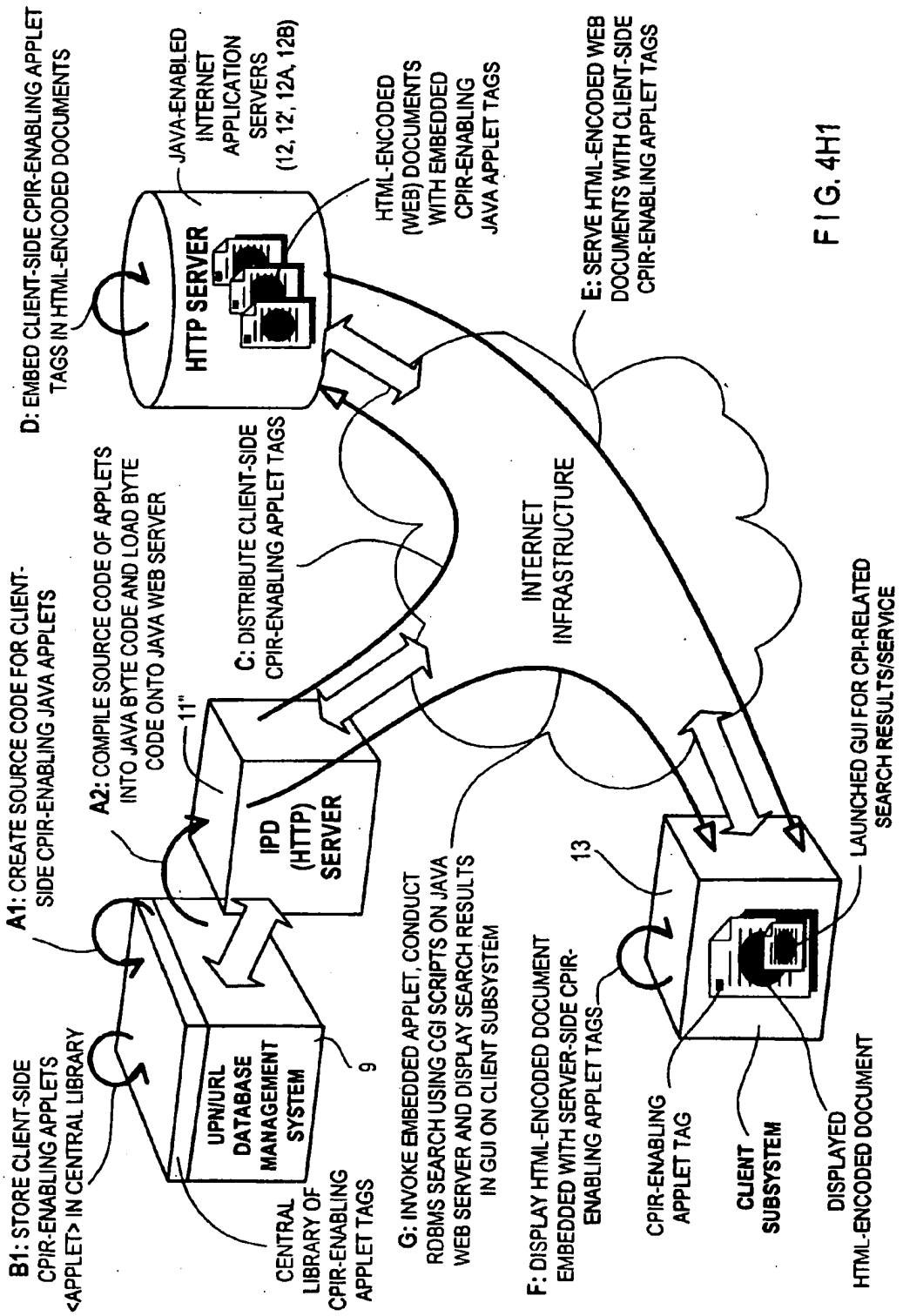
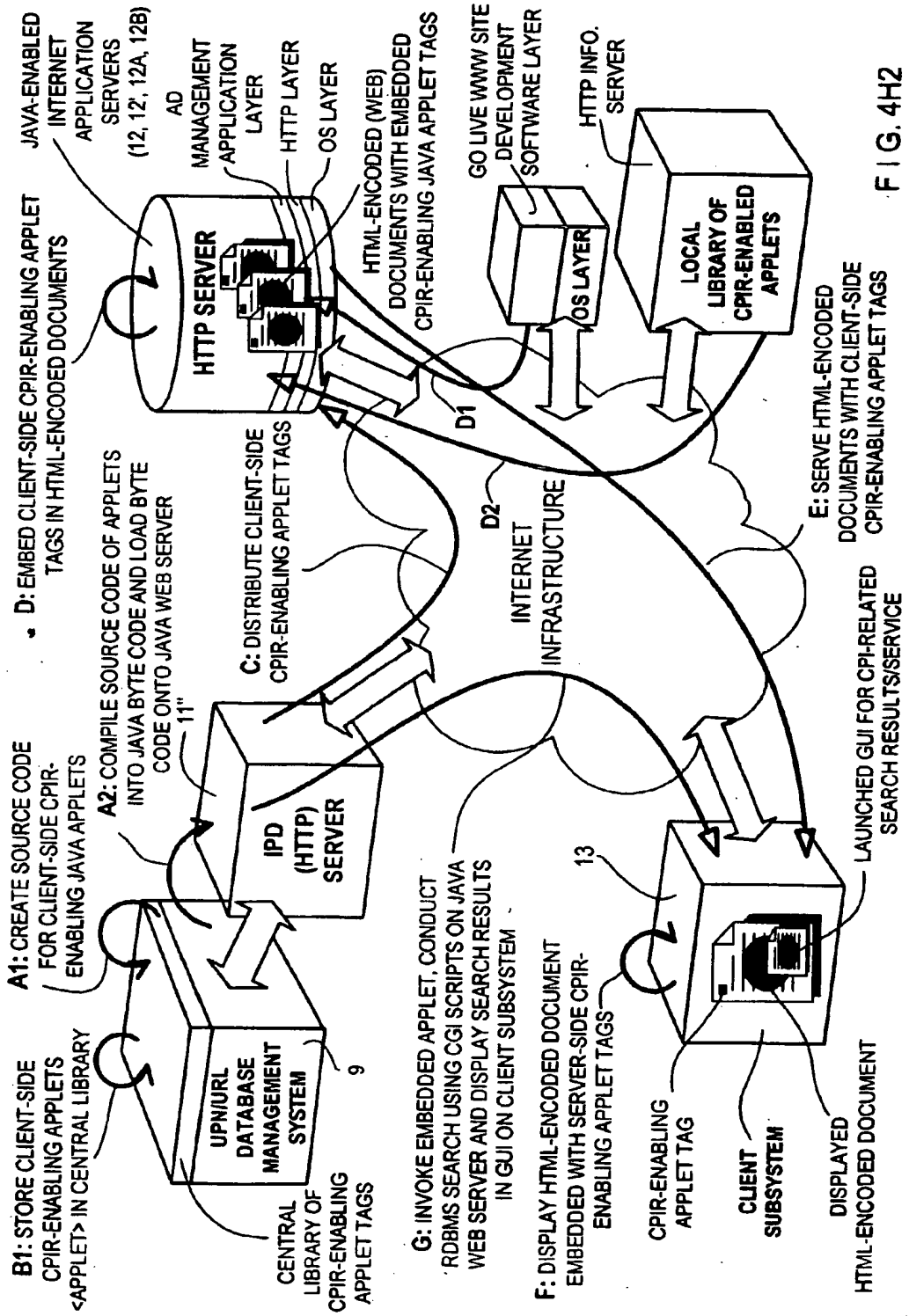

D: EMBED CLIENT-SIDE CPIR-ENABLING APPLET TAGS IN HTML-ENCODED DOCUMENTS
JAVA-ENABLED INTERNET APPLICATION SERVERS (12, 12', 12A, 12B)
HTTP SERVER
HTML-ENCODED (WEB) DOCUMENTS WITH EMBEDDED CPIR-ENABLING JAVA APPLET TAGS
E: SERVE HTML-ENCODED WEB DOCUMENTS WITH CLIENT-SIDE CPIR-ENABLING APPLET TAGS
A2: COMPILE SOURCE CODE OF APPLETS INTO JAVA BYTE CODE AND LOAD BYTE CODE ONTO JAVA WEB SERVER
11"
C: DISTRIBUTE CLIENT-SIDE CPIR-ENABLING APPLET TAGS
INTERNET INFRASTRUCTURE
A1: CREATE SOURCE CODE FOR CLIENT-SIDE CPIR-ENABLING JAVA APPLETS
IPD (HTTP) SERVER
G: INVOKE EMBEDDED APPLET, CONDUCT RDBMS SEARCH USING CGI SCRIPTS ON JAVA WEB SERVER AND DISPLAY SEARCH RESULTS IN GUI ON CLIENT SUBSYSTEM
13
LAUNCHED GUI FOR CPI-RELATED SEARCH RESULTS/SERVICE
F: DISPLAY HTML-ENCODED DOCUMENT EMBEDDED WITH SERVER-SIDE CPIR-ENABLING APPLET TAGS
B1: STORE CLIENT-SIDE CPIR-ENABLING APPLETS <APPLET> IN CENTRAL LIBRARY
UPN/URL DATABASE MANAGEMENT SYSTEM
9
CENTRAL LIBRARY OF CPIR-ENABLING APPLET TAGS
CPIR-ENABLING APPLET TAG
CLIENT SUBSYSTEM
DISPLAYED HTML-ENCODED DOCUMENT

FIG. 4H1


JAVA-ENABLED INTERNET APPLICATION SERVERS (12, 12', 12A, 12B)
AD MANAGEMENT APPLICATION LAYER
HTTP LAYER
OS LAYER
HTTP SERVER
D: EMBED CLIENT-SIDE CPIR-ENABLING APPLET TAGS IN HTML-ENCODED DOCUMENTS
HTML-ENCODED (WEB) DOCUMENTS WITH EMBEDDED CPIR-ENABLING JAVA APPLET TAGS
GO LIVE WWW SITE DEVELOPMENT SOFTWARE LAYER
HTTP INFO. SERVER
OS LAYER
LOCAL LIBRARY OF CPIR-ENABLED APPLETS
D1
D2
INTERNET INFRASTRUCTURE
C: DISTRIBUTE CLIENT-SIDE CPIR-ENABLING APPLET TAGS
A2: COMPILE SOURCE CODE OF APPLETS INTO JAVA BYTE CODE AND LOAD BYTE CODE ONTO JAVA WEB SERVER
11"
IPD (HTTP) SERVER
E: SERVE HTML-ENCODED DOCUMENTS WITH CLIENT-SIDE CPIR-ENABLING APPLET TAGS
A1: CREATE SOURCE CODE FOR CLIENT-SIDE CPIR-ENABLING JAVA APPLETS
G: INVOKE EMBEDDED APPLET, CONDUCT RDBMS SEARCH USING CGI SCRIPTS ON JAVA WEB SERVER AND DISPLAY SEARCH RESULTS IN GUI ON CLIENT SUBSYSTEM
13
F: DISPLAY HTML-ENCODED DOCUMENT EMBEDDED WITH SERVER-SIDE CPIR-ENABLING APPLET TAGS
LAUNCHED GUI FOR CPI-RELATED SEARCH RESULTS/SERVICE
B1: STORE CLIENT-SIDE CPIR-ENABLING APPLETS <APPLET> IN CENTRAL LIBRARY
UPN/URL DATABASE MANAGEMENT SYSTEM
9
CENTRAL LIBRARY OF CPIR-ENABLING APPLET TAGS
CPIR-ENABLING APPLET TAG
CLIENT SUBSYSTEM
DISPLAYED HTML-ENCODED DOCUMENT

F I G. 4H2

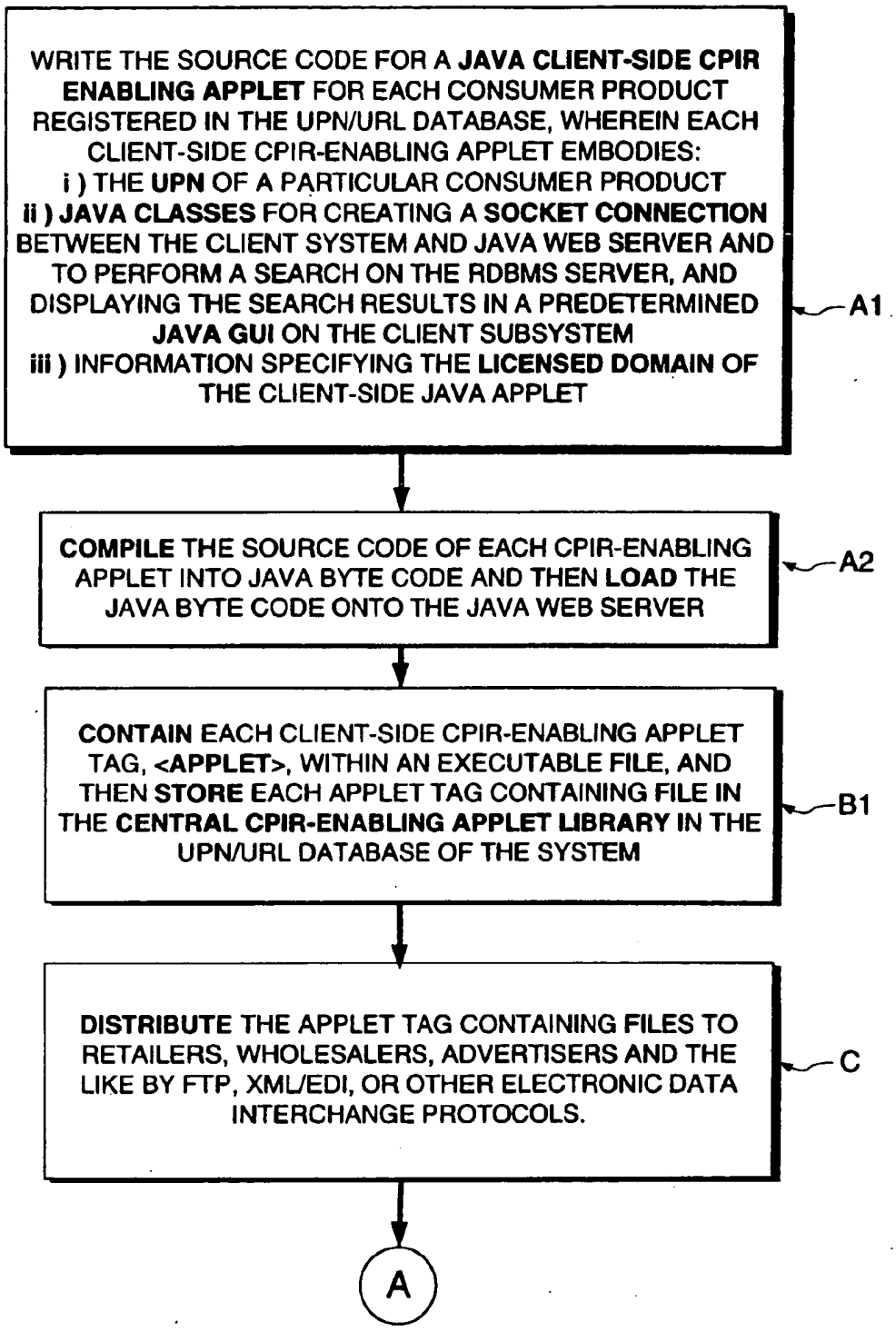

WRITE THE SOURCE CODE FOR A JAVA CLIENT-SIDE CPIR ENABLING APPLET FOR EACH CONSUMER PRODUCT REGISTERED IN THE UPN/URL DATABASE, WHEREIN EACH CLIENT-SIDE CPIR-ENABLING APPLET EMBODIES:
i) THE UPN OF A PARTICULAR CONSUMER PRODUCT
ii) JAVA CLASSES FOR CREATING A SOCKET CONNECTION BETWEEN THE CLIENT SYSTEM AND JAVA WEB SERVER AND TO PERFORM A SEARCH ON THE RDBMS SERVER, AND DISPLAYING THE SEARCH RESULTS IN A PREDETERMINED JAVA GUI ON THE CLIENT SUBSYSTEM
iii) INFORMATION SPECIFYING THE LICENSED DOMAIN OF THE CLIENT-SIDE JAVA APPLET
A1
COMPILE THE SOURCE CODE OF EACH CPIR-ENABLING APPLET INTO JAVA BYTE CODE AND THEN LOAD THE JAVA BYTE CODE ONTO THE JAVA WEB SERVER
A2
CONTAIN EACH CLIENT-SIDE CPIR-ENABLING APPLET TAG, <APPLET>, WITHIN AN EXECUTABLE FILE, AND THEN STORE EACH APPLET TAG CONTAINING FILE IN THE CENTRAL CPIR-ENABLING APPLET LIBRARY IN THE UPN/URL DATABASE OF THE SYSTEM
B1
DISTRIBUTE THE APPLET TAG CONTAINING FILES TO RETAILERS, WHOLESALERS, ADVERTISERS AND THE LIKE BY FTP, XML/EDI, OR OTHER ELECTRONIC DATA INTERCHANGE PROTOCOLS.
C
A

FIG. 4I1

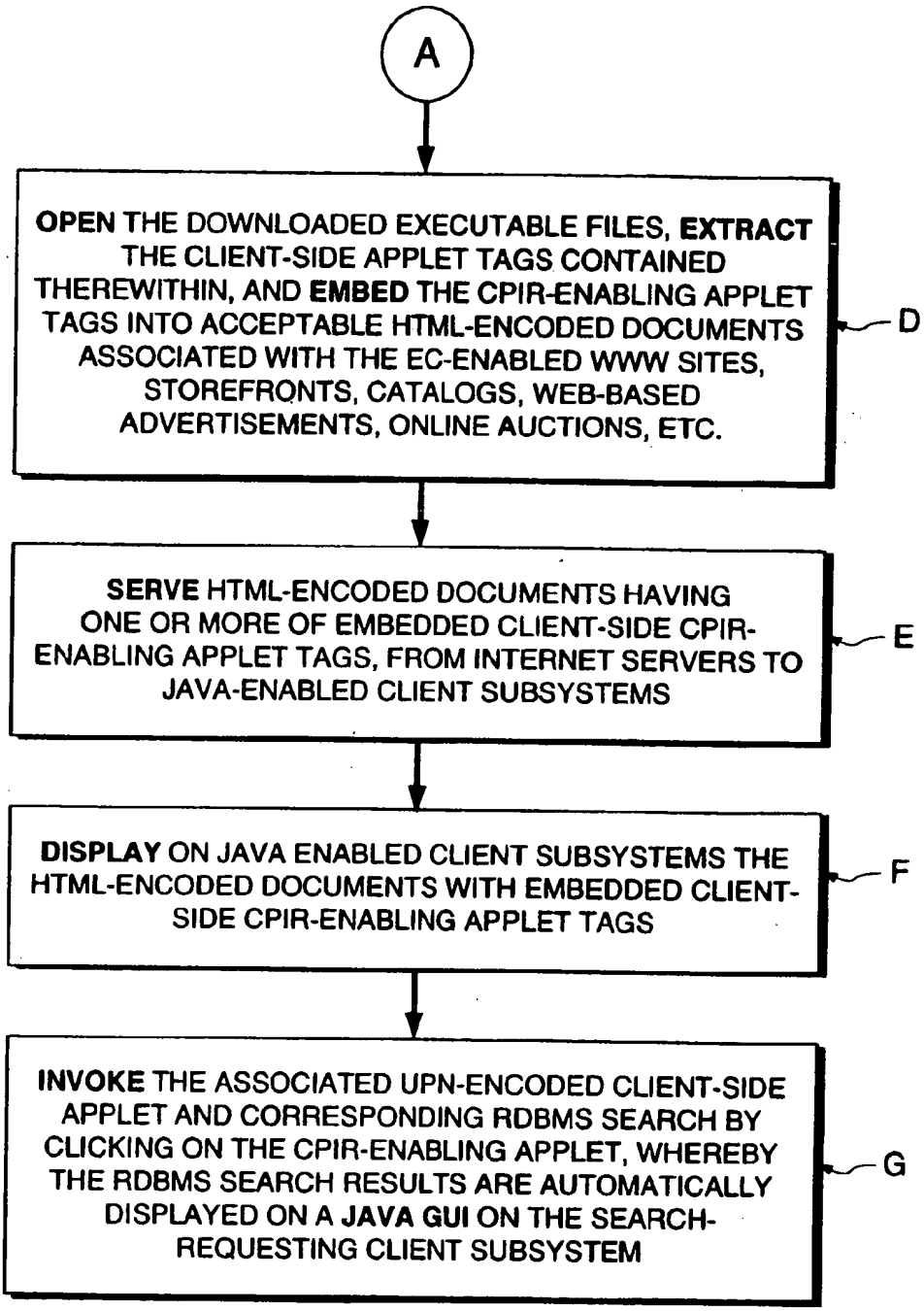

A
OPEN THE DOWNLOADED EXECUTABLE FILES, EXTRACT THE CLIENT-SIDE APPLET TAGS CONTAINED THEREWITHIN, AND EMBED THE CPIR-ENABLING APPLET TAGS INTO ACCEPTABLE HTML-ENCODED DOCUMENTS ASSOCIATED WITH THE EC-ENABLED WWW SITES, STOREFRONTS, CATALOGS, WEB-BASED ADVERTISEMENTS, ONLINE AUCTIONS, ETC.
D
SERVE HTML-ENCODED DOCUMENTS HAVING ONE OR MORE OF EMBEDDED CLIENT-SIDE CPIR-ENABLING APPLET TAGS, FROM INTERNET SERVERS TO JAVA-ENABLED CLIENT SUBSYSTEMS
E
DISPLAY ON JAVA ENABLED CLIENT SUBSYSTEMS THE HTML-ENCODED DOCUMENTS WITH EMBEDDED CLIENT-SIDE CPIR-ENABLING APPLET TAGS
F
INVOKE THE ASSOCIATED UPN-ENCODED CLIENT-SIDE APPLET AND CORRESPONDING RDBMS SEARCH BY CLICKING ON THE CPIR-ENABLING APPLET, WHEREBY THE RDBMS SEARCH RESULTS ARE AUTOMATICALLY DISPLAYED ON A JAVA GUI ON THE SEARCH-REQUESTING CLIENT SUBSYSTEM
G

FIG. 412

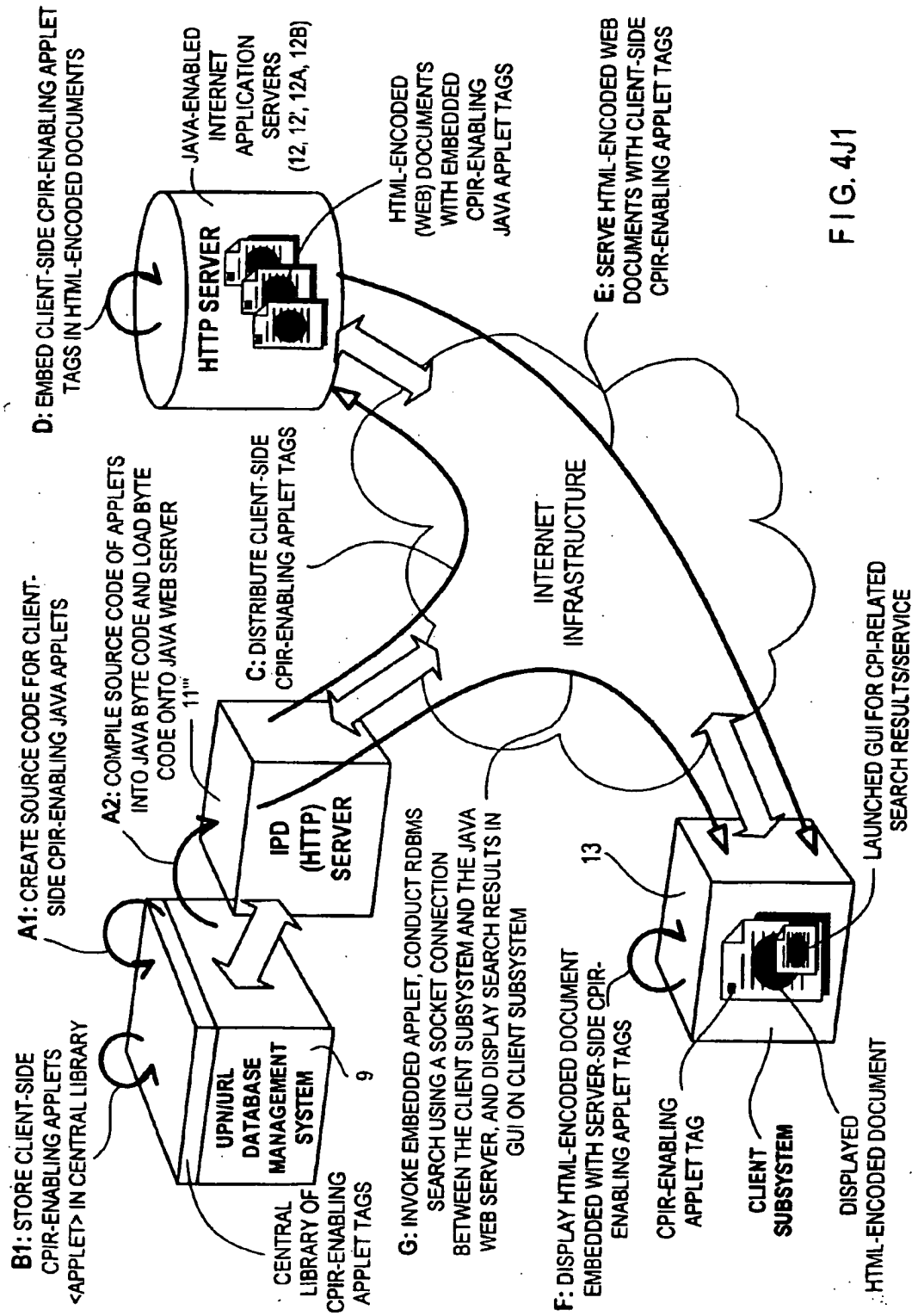

D: EMBED CLIENT-SIDE CPIR-ENABLING APPLET TAGS IN HTML-ENCODED DOCUMENTS
JAVA-ENABLED INTERNET APPLICATION SERVERS (12, 12', 12A, 12B)
HTTP SERVER
HTML-ENCODED (WEB) DOCUMENTS WITH EMBEDDED CPIR-ENABLING JAVA APPLET TAGS
E: SERVE HTML-ENCODED WEB DOCUMENTS WITH CLIENT-SIDE CPIR-ENABLING APPLET TAGS
A1: CREATE SOURCE CODE FOR CLIENT-SIDE CPIR-ENABLING JAVA APPLETS
A2: COMPILE SOURCE CODE OF APPLETS INTO JAVA BYTE CODE AND LOAD BYTE CODE ONTO JAVA WEB SERVER
11''
C: DISTRIBUTE CLIENT-SIDE CPIR-ENABLING APPLET TAGS
INTERNET INFRASTRUCTURE
IPD (HTTP) SERVER
G: INVOKE EMBEDDED APPLET, CONDUCT RDBMS SEARCH USING A SOCKET CONNECTION BETWEEN THE CLIENT SUBSYSTEM AND THE JAVA WEB SERVER, AND DISPLAY SEARCH RESULTS IN GUI ON CLIENT SUBSYSTEM
13
F: DISPLAY HTML-ENCODED DOCUMENT EMBEDDED WITH SERVER-SIDE CPIR-ENABLING APPLET TAGS
LAUNCHED GUI FOR CPI-RELATED SEARCH RESULTS/SERVICE
B1: STORE CLIENT-SIDE CPIR-ENABLING APPLETS <APPLET> IN CENTRAL LIBRARY
UPN/URL DATABASE MANAGEMENT SYSTEM
9
CENTRAL LIBRARY OF CPIR-ENABLING APPLET TAGS
CPIR-ENABLING APPLET TAG
CLIENT SUBSYSTEM
DISPLAYED HTML-ENCODED DOCUMENT

FIG. 4J1

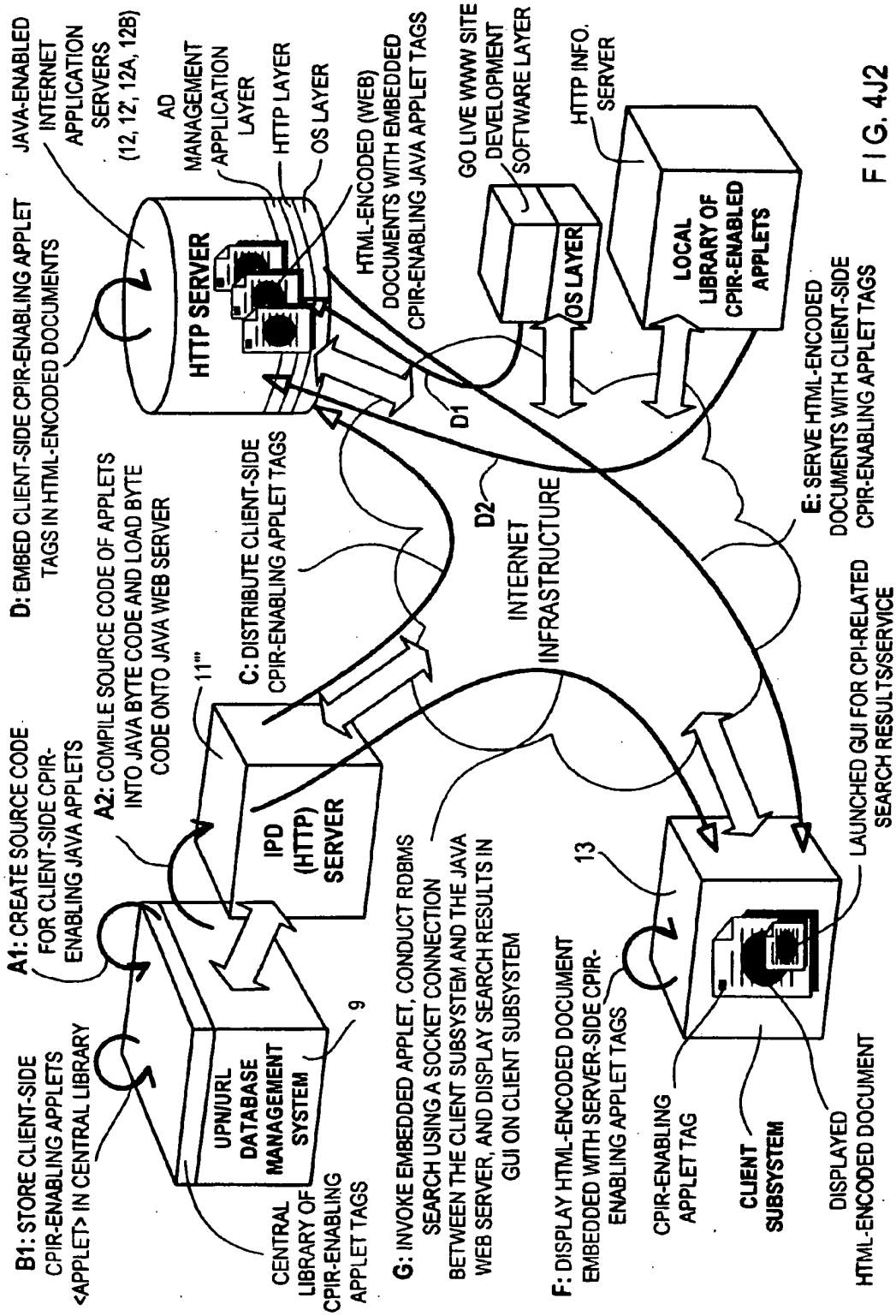

JAVA-ENABLED INTERNET APPLICATION SERVERS (12, 12', 12A, 12B)
AD MANAGEMENT APPLICATION LAYER
HTTP LAYER
OS LAYER
HTML-ENCODED (WEB) DOCUMENTS WITH EMBEDDED CPIR-ENABLING JAVA APPLET TAGS
GO LIVE WWW SITE DEVELOPMENT SOFTWARE LAYER
HTTP INFO. SERVER
HTTP SERVER
OS LAYER
LOCAL LIBRARY OF CPIR-ENABLED APPLETS
D: EMBED CLIENT-SIDE CPIR-ENABLING APPLET TAGS IN HTML-ENCODED DOCUMENTS
D1
D2
INTERNET INFRASTRUCTURE
E: SERVE HTML-ENCODED DOCUMENTS WITH CLIENT-SIDE CPIR-ENABLING APPLET TAGS
A1: CREATE SOURCE CODE FOR CLIENT-SIDE CPIR-ENABLING JAVA APPLETS
A2: COMPILE SOURCE CODE OF APPLETS INTO JAVA BYTE CODE AND LOAD BYTE CODE ONTO JAVA WEB SERVER
11'''
C: DISTRIBUTE CLIENT-SIDE CPIR-ENABLING APPLET TAGS
IPD (HTTP) SERVER
B1: STORE CLIENT-SIDE CPIR-ENABLING APPLETS <APPLET> IN CENTRAL LIBRARY
UPN/URL DATABASE MANAGEMENT SYSTEM
9
CENTRAL LIBRARY OF CPIR-ENABLING APPLET TAGS
G: INVOKE EMBEDDED APPLET, CONDUCT RDBMS SEARCH USING A SOCKET CONNECTION BETWEEN THE CLIENT SUBSYSTEM AND THE JAVA WEB SERVER, AND DISPLAY SEARCH RESULTS IN GUI ON CLIENT SUBSYSTEM
13
F: DISPLAY HTML-ENCODED DOCUMENT EMBEDDED WITH SERVER-SIDE CPIR-ENABLING APPLET TAGS
CPIR-ENABLING APPLET TAG
CLIENT SUBSYSTEM
DISPLAYED HTML-ENCODED DOCUMENT
LAUNCHED GUI FOR CPI-RELATED SEARCH RESULTS/SERVICE

FIG. 4J2

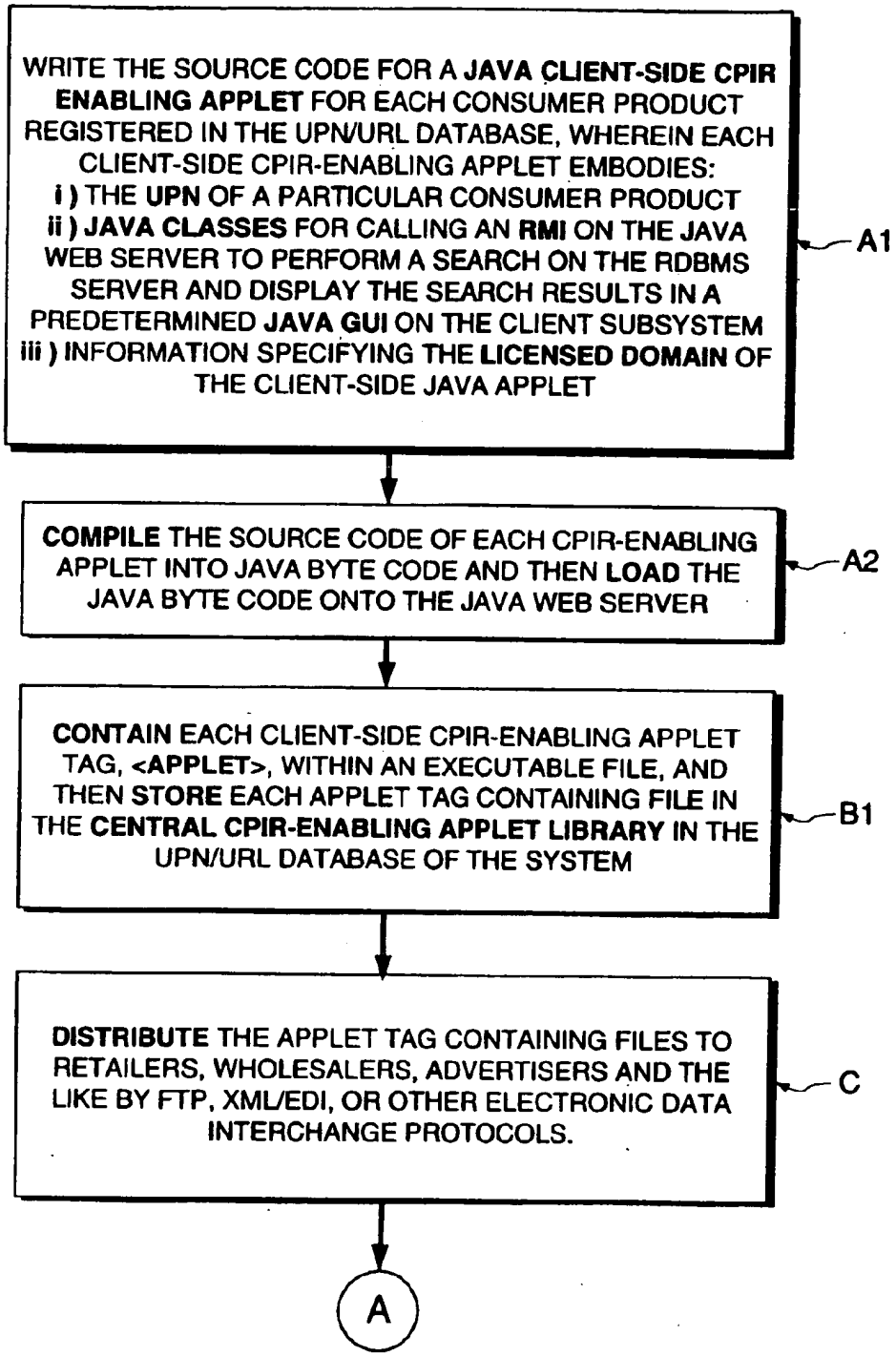

WRITE THE SOURCE CODE FOR A JAVA CLIENT-SIDE CPIR ENABLING APPLET FOR EACH CONSUMER PRODUCT REGISTERED IN THE UPN/URL DATABASE, WHEREIN EACH CLIENT-SIDE CPIR-ENABLING APPLET EMBODIES:
i) THE UPN OF A PARTICULAR CONSUMER PRODUCT
ii) JAVA CLASSES FOR CALLING AN RMI ON THE JAVA WEB SERVER TO PERFORM A SEARCH ON THE RDBMS SERVER AND DISPLAY THE SEARCH RESULTS IN A PREDETERMINED JAVA GUI ON THE CLIENT SUBSYSTEM
iii) INFORMATION SPECIFYING THE LICENSED DOMAIN OF THE CLIENT-SIDE JAVA APPLET
A1
COMPILE THE SOURCE CODE OF EACH CPIR-ENABLING APPLET INTO JAVA BYTE CODE AND THEN LOAD THE JAVA BYTE CODE ONTO THE JAVA WEB SERVER
A2
CONTAIN EACH CLIENT-SIDE CPIR-ENABLING APPLET TAG, <APPLET>, WITHIN AN EXECUTABLE FILE, AND THEN STORE EACH APPLET TAG CONTAINING FILE IN THE CENTRAL CPIR-ENABLING APPLET LIBRARY IN THE UPN/URL DATABASE OF THE SYSTEM
B1
DISTRIBUTE THE APPLET TAG CONTAINING FILES TO RETAILERS, WHOLESALERS, ADVERTISERS AND THE LIKE BY FTP, XML/EDI, OR OTHER ELECTRONIC DATA INTERCHANGE PROTOCOLS.
C
A

FIG. 4K1

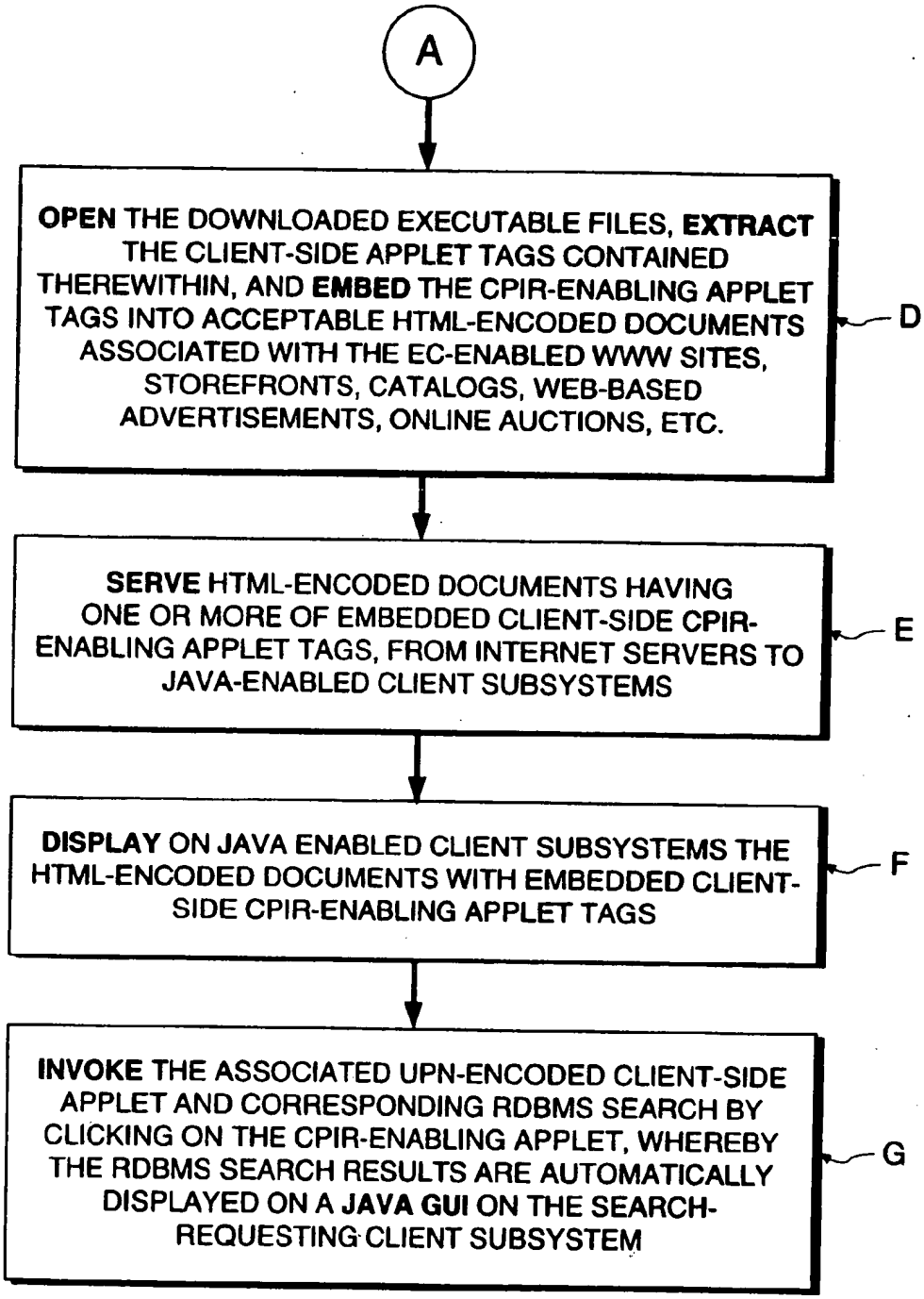

A
OPEN THE DOWNLOADED EXECUTABLE FILES, EXTRACT THE CLIENT-SIDE APPLET TAGS CONTAINED THEREWITHIN, AND EMBED THE CPIR-ENABLING APPLET TAGS INTO ACCEPTABLE HTML-ENCODED DOCUMENTS ASSOCIATED WITH THE EC-ENABLED WWW SITES, STOREFRONTS, CATALOGS, WEB-BASED ADVERTISEMENTS, ONLINE AUCTIONS, ETC.
D
SERVE HTML-ENCODED DOCUMENTS HAVING ONE OR MORE OF EMBEDDED CLIENT-SIDE CPIR-ENABLING APPLET TAGS, FROM INTERNET SERVERS TO JAVA-ENABLED CLIENT SUBSYSTEMS
E
DISPLAY ON JAVA ENABLED CLIENT SUBSYSTEMS THE HTML-ENCODED DOCUMENTS WITH EMBEDDED CLIENT-SIDE CPIR-ENABLING APPLET TAGS
F
INVOKE THE ASSOCIATED UPN-ENCODED CLIENT-SIDE APPLET AND CORRESPONDING RDBMS SEARCH BY CLICKING ON THE CPIR-ENABLING APPLET, WHEREBY THE RDBMS SEARCH RESULTS ARE AUTOMATICALLY DISPLAYED ON A JAVA GUI ON THE SEARCH-REQUESTING CLIENT SUBSYSTEM
G

FIG. 4K2

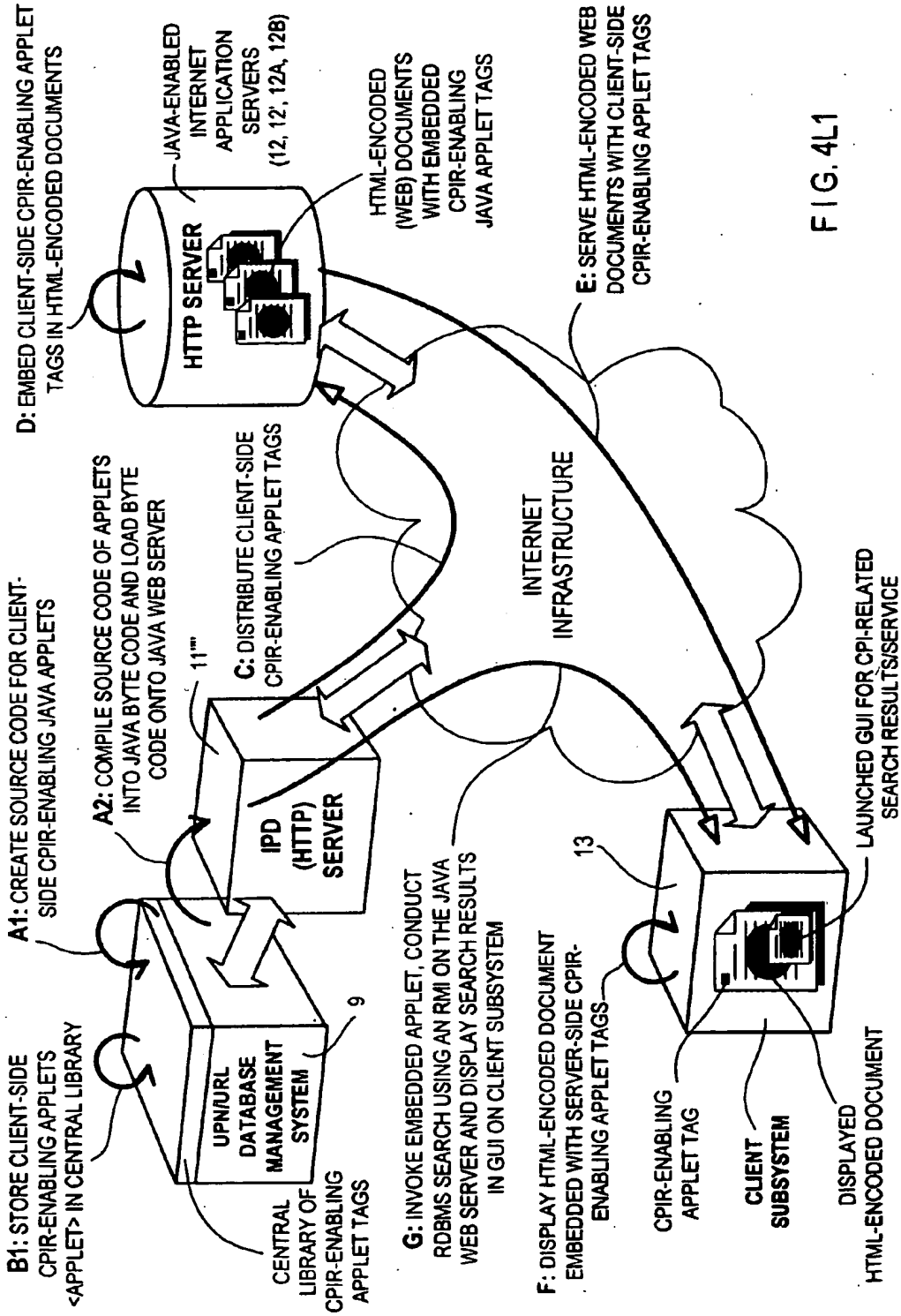

D: EMBED CLIENT-SIDE CPIR-ENABLING APPLET TAGS IN HTML-ENCODED DOCUMENTS
JAVA-ENABLED INTERNET APPLICATION SERVERS (12, 12', 12A, 12B)
HTTP SERVER
HTML-ENCODED (WEB) DOCUMENTS WITH EMBEDDED CPIR-ENABLING JAVA APPLET TAGS
E: SERVE HTML-ENCODED WEB DOCUMENTS WITH CLIENT-SIDE CPIR-ENABLING APPLET TAGS
A1: CREATE SOURCE CODE FOR CLIENT-SIDE CPIR-ENABLING JAVA APPLETS
A2: COMPILE SOURCE CODE OF APPLETS INTO JAVA BYTE CODE AND LOAD BYTE CODE ONTO JAVA WEB SERVER
11""
C: DISTRIBUTE CLIENT-SIDE CPIR-ENABLING APPLET TAGS
INTERNET INFRASTRUCTURE
IPD (HTTP) SERVER
B1: STORE CLIENT-SIDE CPIR-ENABLING APPLETS <APPLET> IN CENTRAL LIBRARY
UPN/URL DATABASE MANAGEMENT SYSTEM
9
CENTRAL LIBRARY OF CPIR-ENABLING APPLET TAGS
G: INVOKE EMBEDDED APPLET, CONDUCT RDBMS SEARCH USING AN RMI ON THE JAVA WEB SERVER AND DISPLAY SEARCH RESULTS IN GUI ON CLIENT SUBSYSTEM
13
F: DISPLAY HTML-ENCODED DOCUMENT EMBEDDED WITH SERVER-SIDE CPIR-ENABLING APPLET TAGS
LAUNCHED GUI FOR CPI-RELATED SEARCH RESULTS/SERVICE
CPIR-ENABLING APPLET TAG
CLIENT SUBSYSTEM
DISPLAYED HTML-ENCODED DOCUMENT

FIG. 4L1

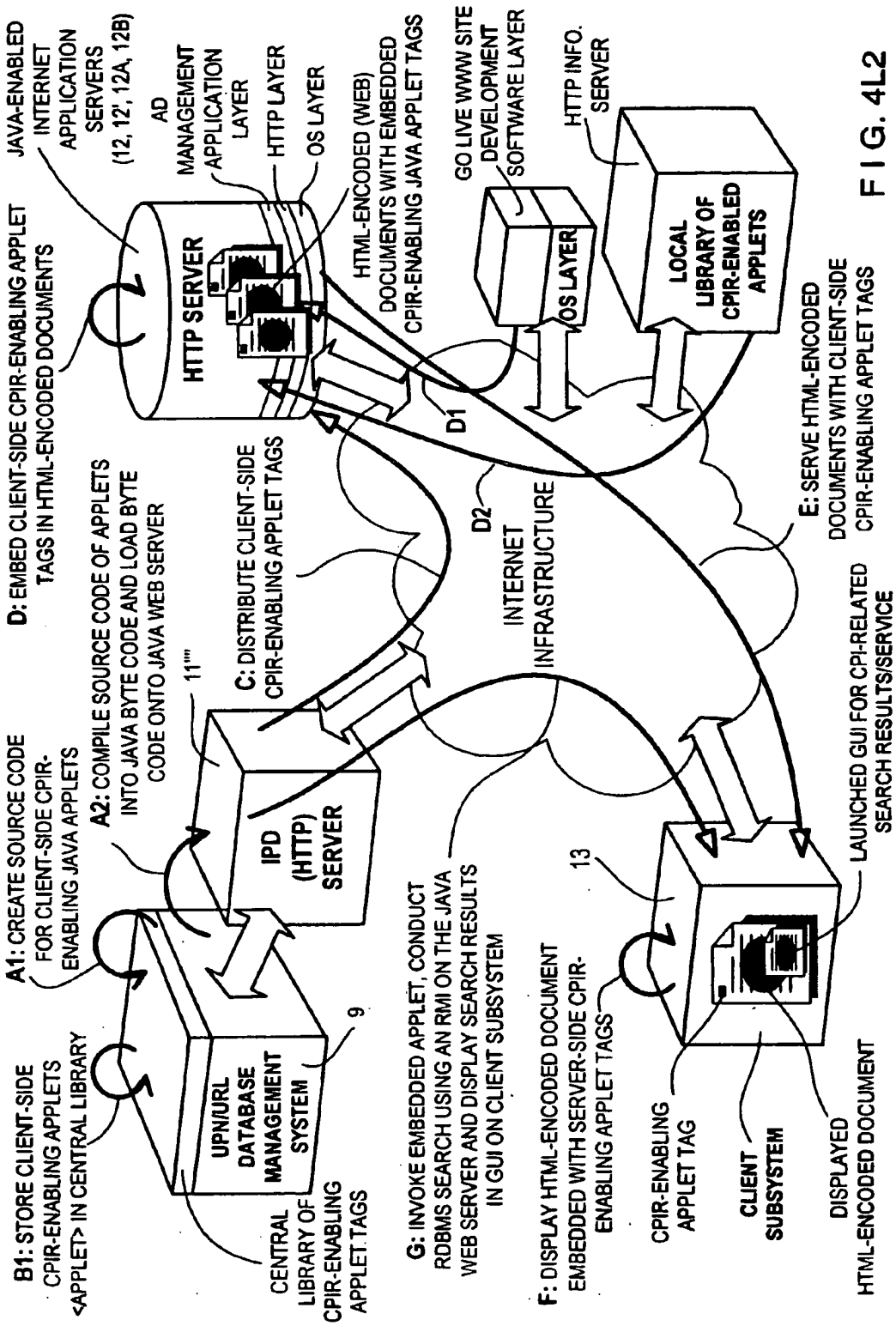

JAVA-ENABLED INTERNET APPLICATION SERVERS (12, 12', 12A, 12B)
AD MANAGEMENT APPLICATION LAYER
HTTP LAYER
OS LAYER
HTTP SERVER
D: EMBED CLIENT-SIDE CPIR-ENABLING APPLET TAGS IN HTML-ENCODED DOCUMENTS
HTML-ENCODED (WEB) DOCUMENTS WITH EMBEDDED CPIR-ENABLING JAVA APPLET TAGS
GO LIVE WWW SITE DEVELOPMENT SOFTWARE LAYER
HTTP INFO. SERVER
OS LAYER
LOCAL LIBRARY OF CPIR-ENABLED APPLETS
D1
D2
INTERNET INFRASTRUCTURE
E: SERVE HTML-ENCODED DOCUMENTS WITH CLIENT-SIDE CPIR-ENABLING APPLET TAGS
LAUNCHED GUI FOR CPI-RELATED SEARCH RESULTS/SERVICE
A1: CREATE SOURCE CODE FOR CLIENT-SIDE CPIR-ENABLING JAVA APPLETS
A2: COMPILE SOURCE CODE OF APPLETS INTO JAVA BYTE CODE AND LOAD BYTE CODE ONTO JAVA WEB SERVER
11''''
C: DISTRIBUTE CLIENT-SIDE CPIR-ENABLING APPLET TAGS
IPD (HTTP) SERVER
G: INVOKE EMBEDDED APPLET, CONDUCT RDBMS SEARCH USING AN RMI ON THE JAVA WEB SERVER AND DISPLAY SEARCH RESULTS IN GUI ON CLIENT SUBSYSTEM
13
F: DISPLAY HTML-ENCODED DOCUMENT EMBEDDED WITH SERVER-SIDE CPIR-ENABLING APPLET TAGS
CPIR-ENABLING APPLET TAG
CLIENT SUBSYSTEM
DISPLAYED HTML-ENCODED DOCUMENT
B1: STORE CLIENT-SIDE CPIR-ENABLING APPLETS <APPLET> IN CENTRAL LIBRARY
UPN/URL DATABASE MANAGEMENT SYSTEM
9
CENTRAL LIBRARY OF CPIR-ENABLING APPLET TAGS

FIG. 4L2

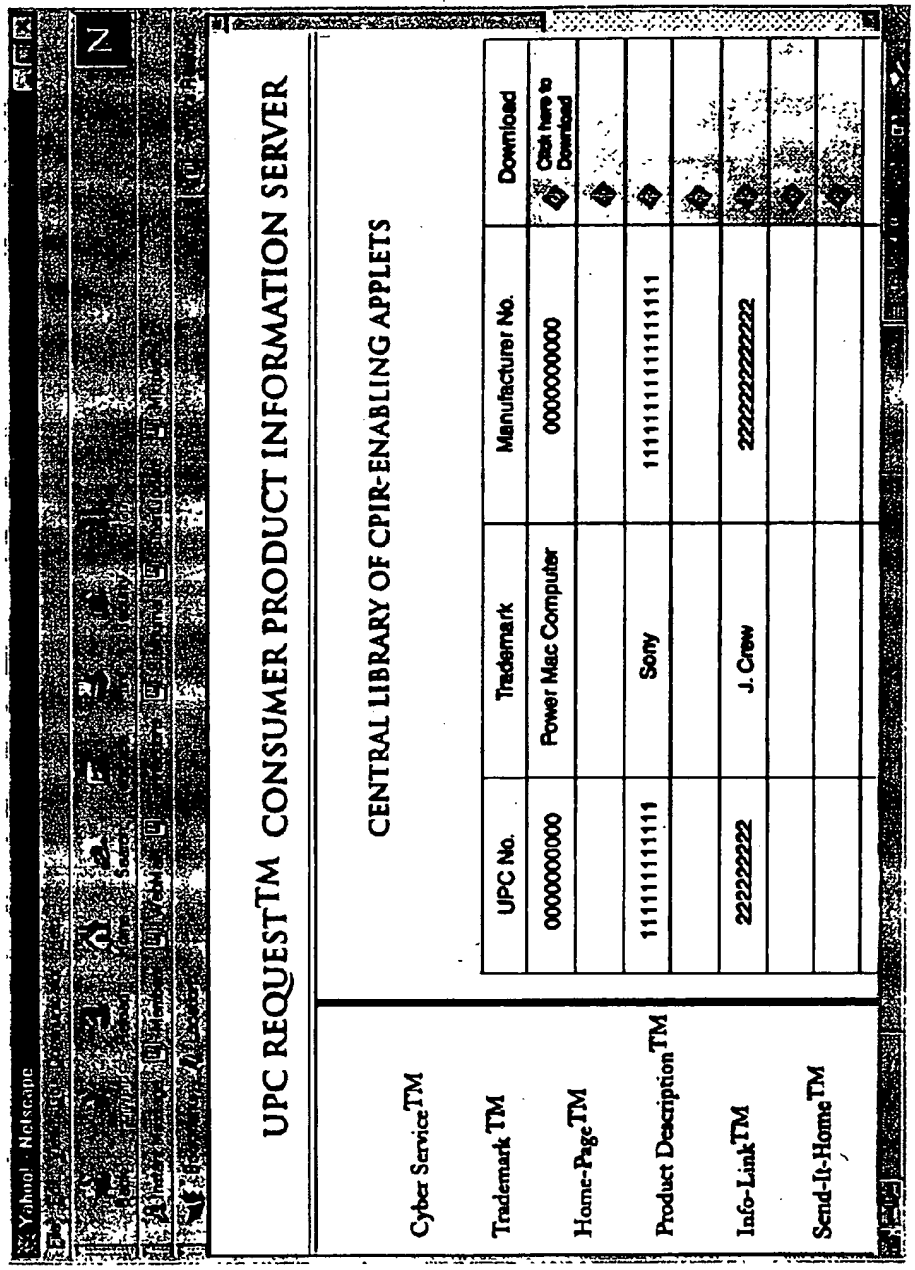

Yahoo! - Netscape
UPC REQUEST™ CONSUMER PRODUCT INFORMATION SERVER
CENTRAL LIBRARY OF CPIR-ENABLING APPLETS
Cyber Service™
Trademark™
Home-Page™
Product Description™
Info-Link™
Send-It-Home™
UPC No.
Trademark
Manufacturer No.
Download
0000000000
Power Mac Computer
0000000000
Click here to Download
11111111111
Sony
11111111111111111
22222222
J. Crew
22222222222222

FIG. 4M1

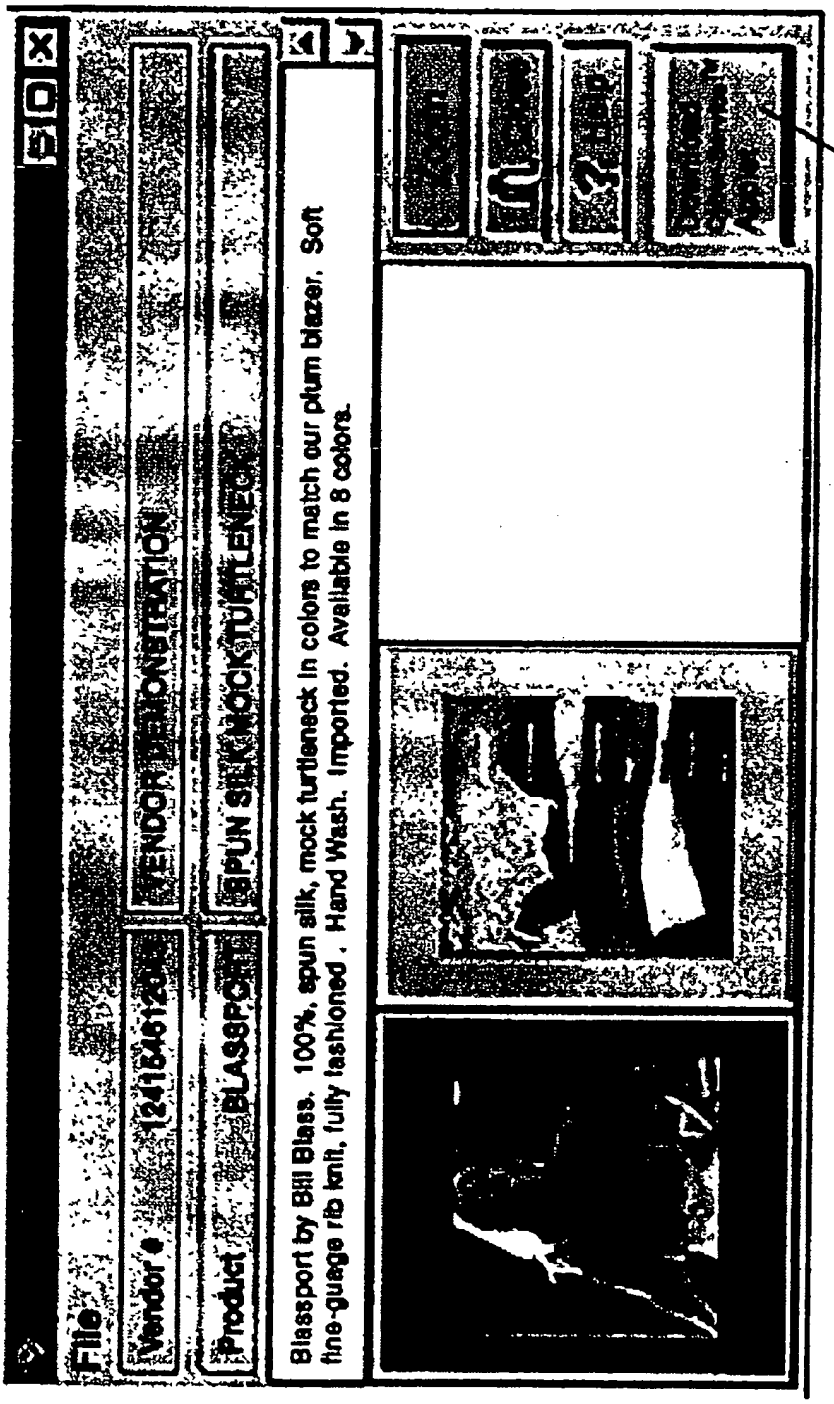

File
Vendor
VENDOR DEMONSTRATION
Product
SPUN SILK MOCK TURTLENECK
Blassport by Bill Blass. 100%, spun silk, mock turtleneck in colors to match our plum blazer. Soft fine-guage rib knit, fully fashioned . Hand Wash. Imported. Available in 8 colors.
100

FIG. 4M2

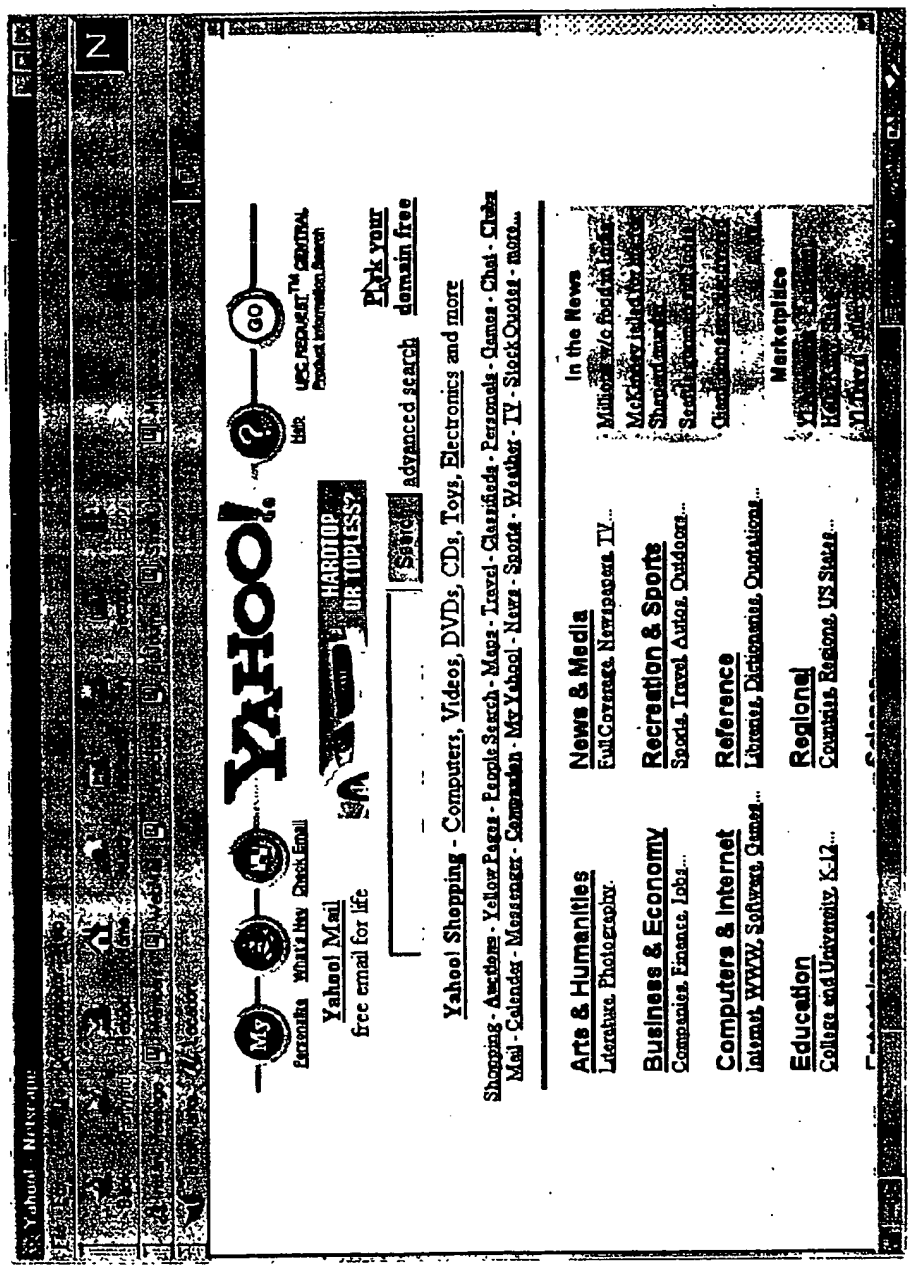

YAHOO!
GO
Yahoo! Mail
free email for life
HARDTOP OR TOPLESS?
Search
advanced search
Pick your domain free
Yahoo! Shopping - Computers, Videos, DVDs, CDs, Toys, Electronics and more
Shopping - Auctions - Yellow Pages - People Search - Maps - Travel - Classifieds - Personals - Games - Chat - Clubs
Mail - Calendar - Messenger - Companion - My Yahoo! - News - Sports - Weather - TV - StockQuotes - more...
Arts & Humanities
Literature, Photography...
News & Media
Full Coverage, Newspapers, TV...
In the News
Business & Economy
Companies, Finance, Jobs...
Recreation & Sports
Sports, Travel, Autos, Outdoors...
Computers & Internet
Internet, WWW, Software, Games...
Reference
Libraries, Dictionaries, Quotations...
Marketplace
Education
College and University, K-12...
Regional
Countries, Regions, US States...

FIG. 4N1

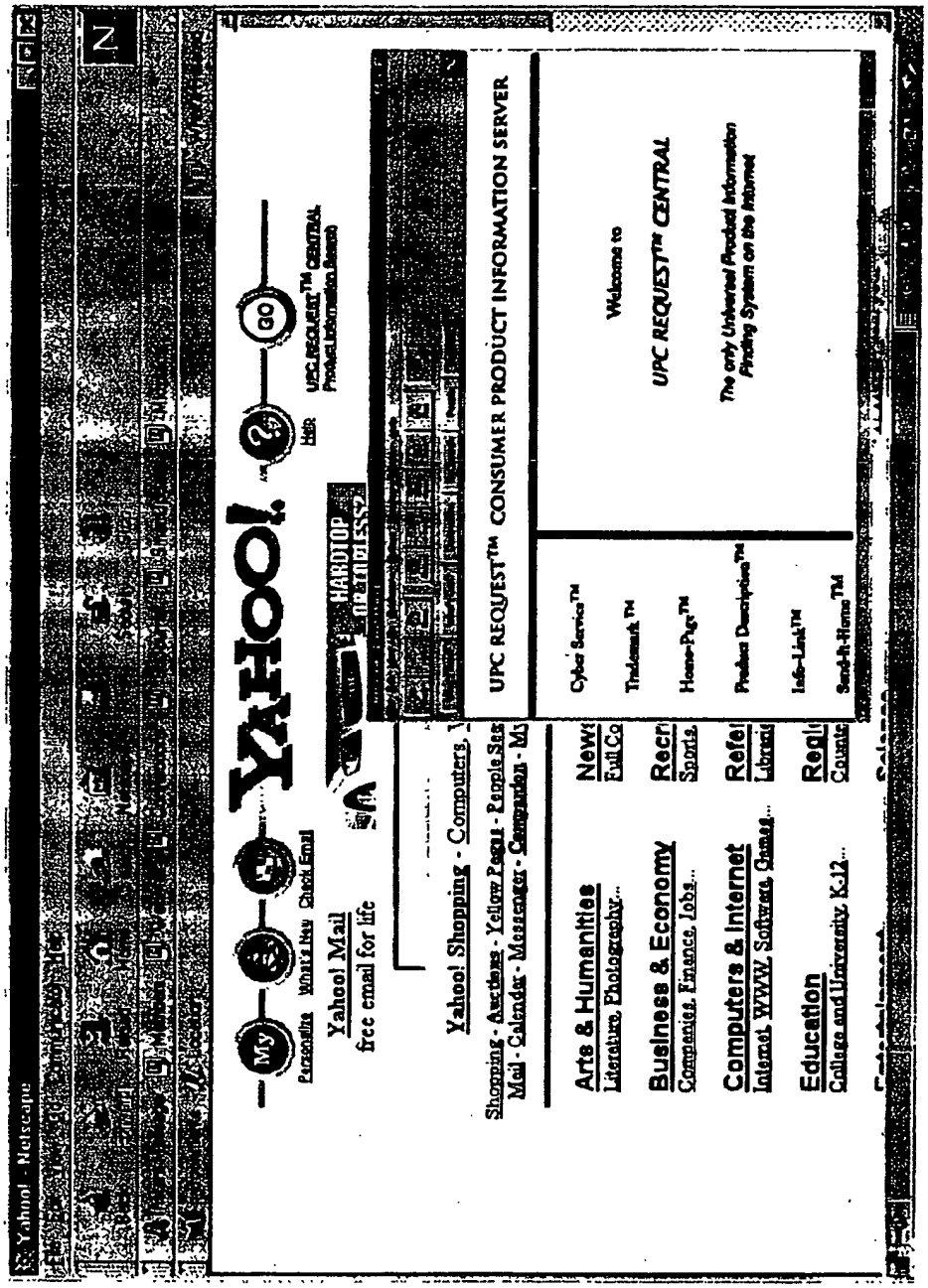

Yahoo! - Netscape
YAHOO!
Yahoo! Mail
free email for life
Yahoo! Shopping - Computers
UPC REQUEST™ CONSUMER PRODUCT INFORMATION SERVER
Welcome to
UPC REQUEST™ CENTRAL
The only Universal Product Information Finding System on the Internet
Arts & Humanities
Literature, Photography...
Business & Economy
Companies, Finance, Jobs...
Computers & Internet
Internet, WWW, Software, Games...
Education
College and University, K-12...

FIG. 4N2

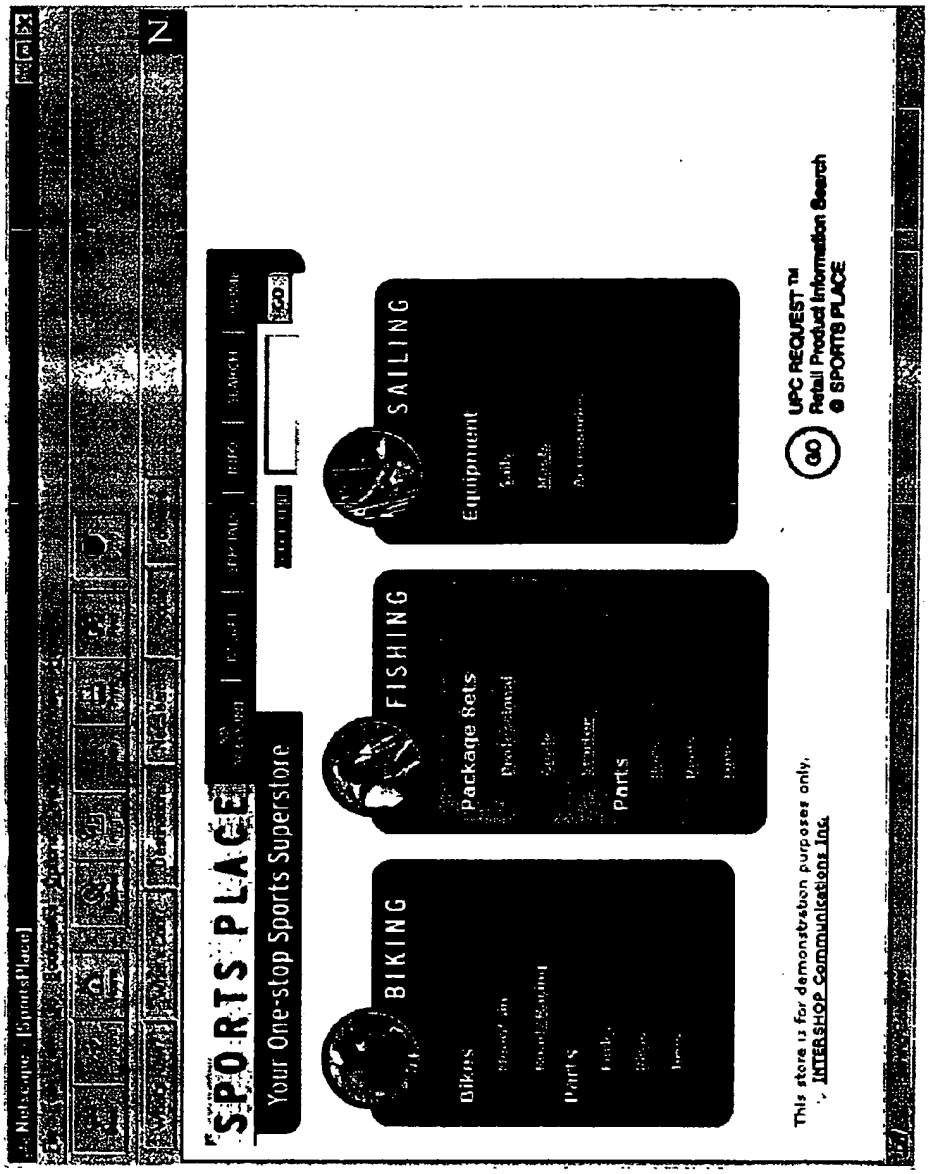

SPORTS PLACE
Your One-stop Sports Superstore
BIKING
Bikes
Parts
FISHING
Package Sets
Parts
SAILING
Equipment
This store is for demonstration purposes only.
INTERSHOP Communications Inc.
UPC REQUEST ™
Retail Product Information Search
@ SPORTS PLACE

FIG. 401

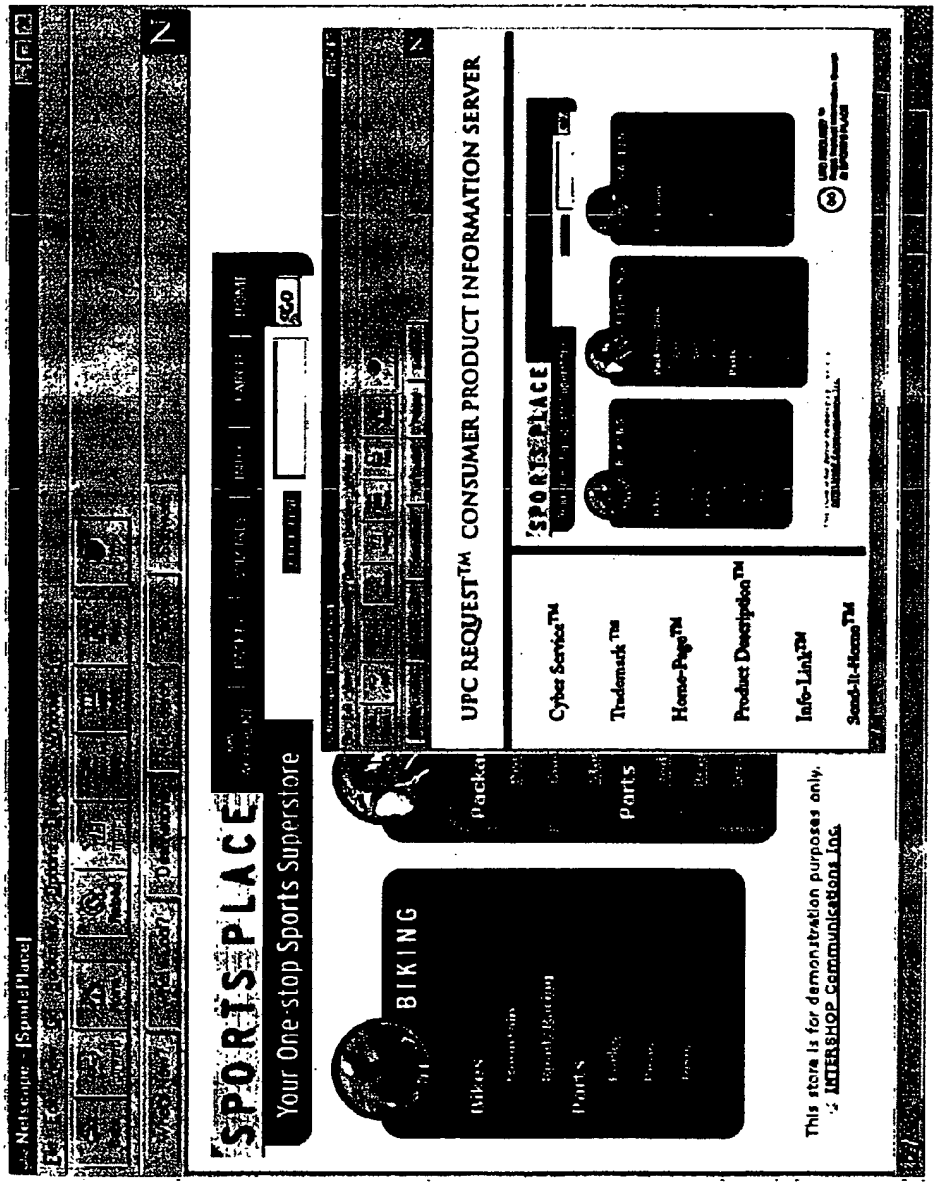

SPORTSPLACE
Your One-stop Sports Superstore
GO
BIKING
Parts
UPC REQUEST™ CONSUMER PRODUCT INFORMATION SERVER
Cyber Service™
Trademark™
Home-Page™
Product Description™
Info-Link™
Send-It-Home™
This store is for demonstration purposes only.
INTERSHOP Communications Inc.

FIG. 402

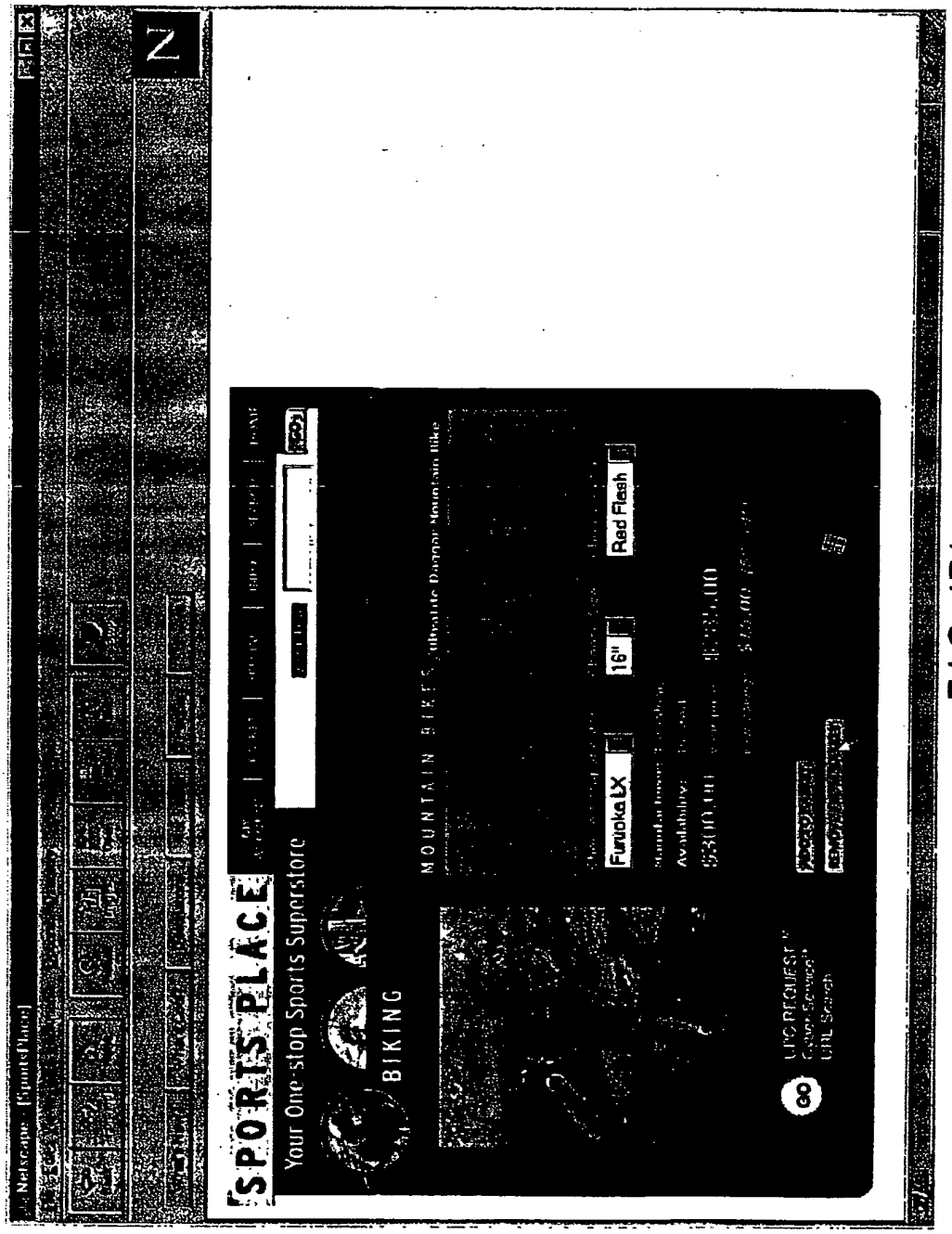

SPORTS PLACE
Your One-stop Sports Superstore
BIKING
MOUNTAIN BIKES
Furtoka LX
16"
Red Flash

FIG. 4P1

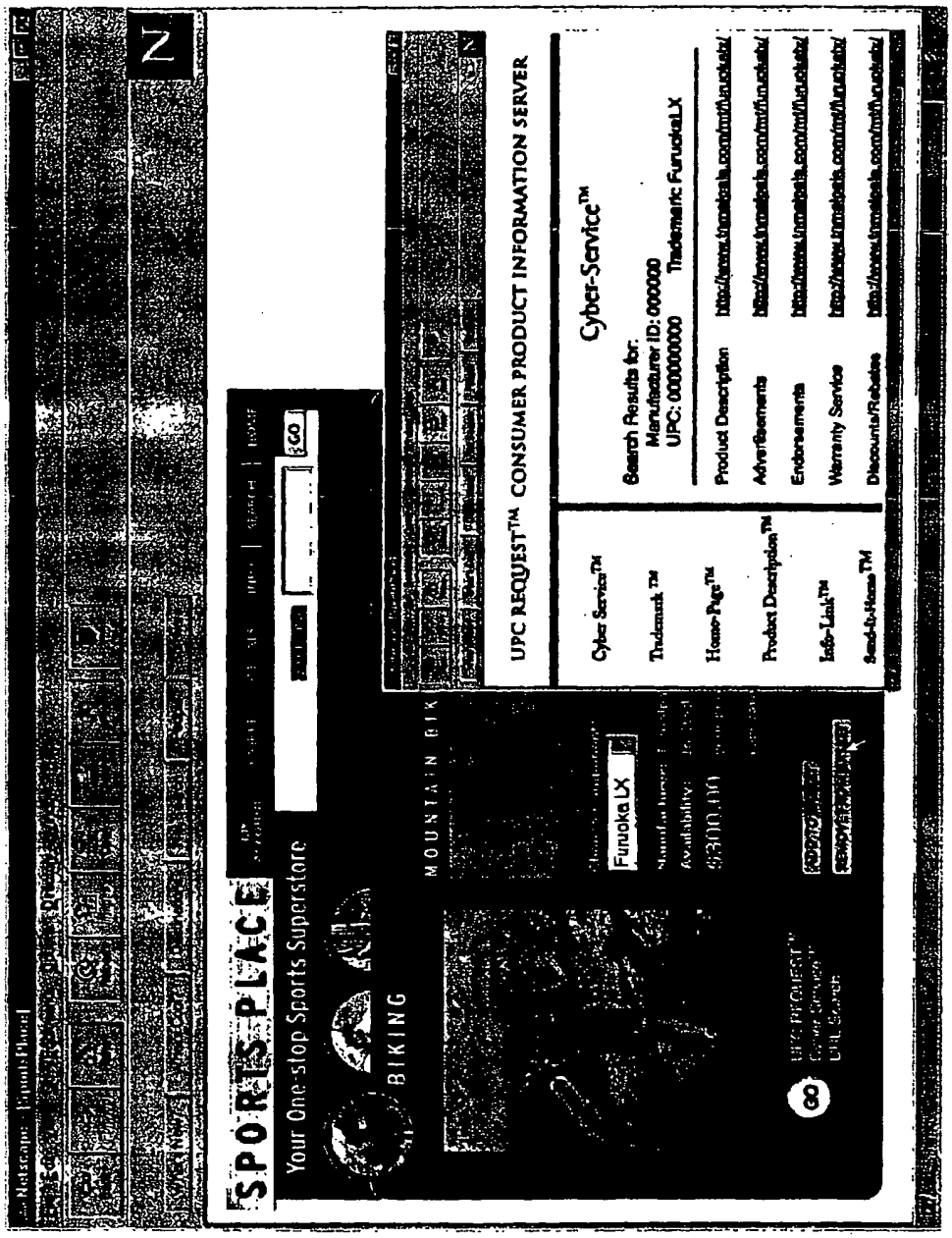

SPORTS PLACE
Your One-stop Sports Superstore
BIKING
MOUNTAIN BIK
Funuoka LX
UPC REQUEST™ CONSUMER PRODUCT INFORMATION SERVER
Cyber-Service™
Search Results for:
Manufacturer ID: 000000
UPC: 000000000 Trademark: FunuokaLX
Cyber Service™
Trademark ™
Home-Page™
Product Description™
Info-Link™
Send-to-Home™
Product Description
Advertisements
Endorsements
Warranty Service
Discounts/Rebates

FIG. 4P2

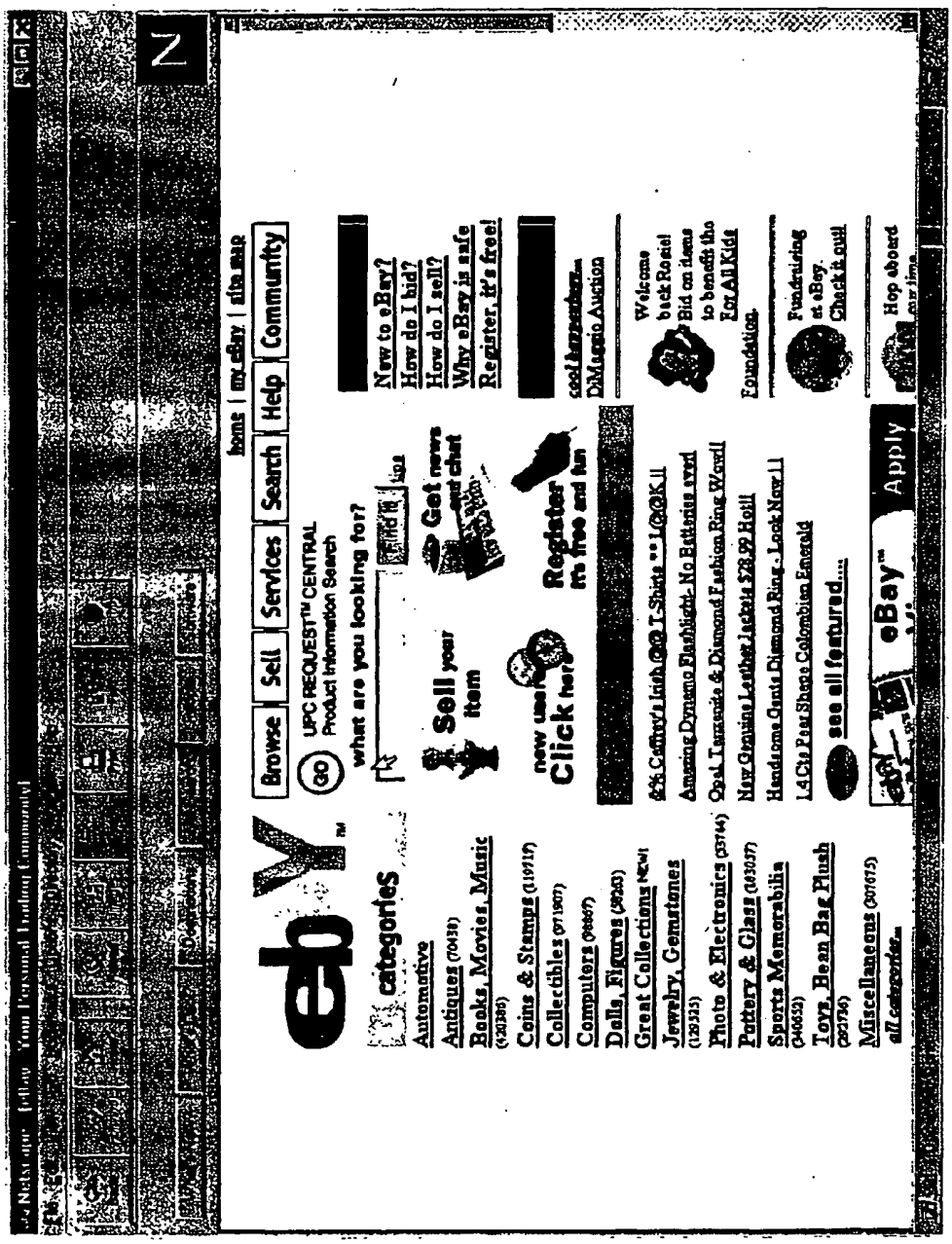

Browse
Sell
Services
Search
Help
Community
UPC REQUEST™ CENTRAL
Product Information Search
what are you looking for?
Sell your item
Get news and chat
Register
It's free and fun
categories
Automotive
Antiques
Books, Movies, Music
Coins & Stamps
Collectibles
Computers
Dolls, Figures
Great Collections
Jewelry, Gemstones
Photo & Electronics
Pottery & Glass
Sports Memorabilia
Toys, Bean Bag Plush
Miscellaneous
New to eBay?
How do I bid?
How do I sell?
Why eBay is safe
Register, it's free!
Welcome back Rosie!
Bid on items to benefit the Ear All Kids Foundation.
Fundraising at eBay. Check it out!
Hop aboard our ...
eBay
Apply

FIG. 4Q1

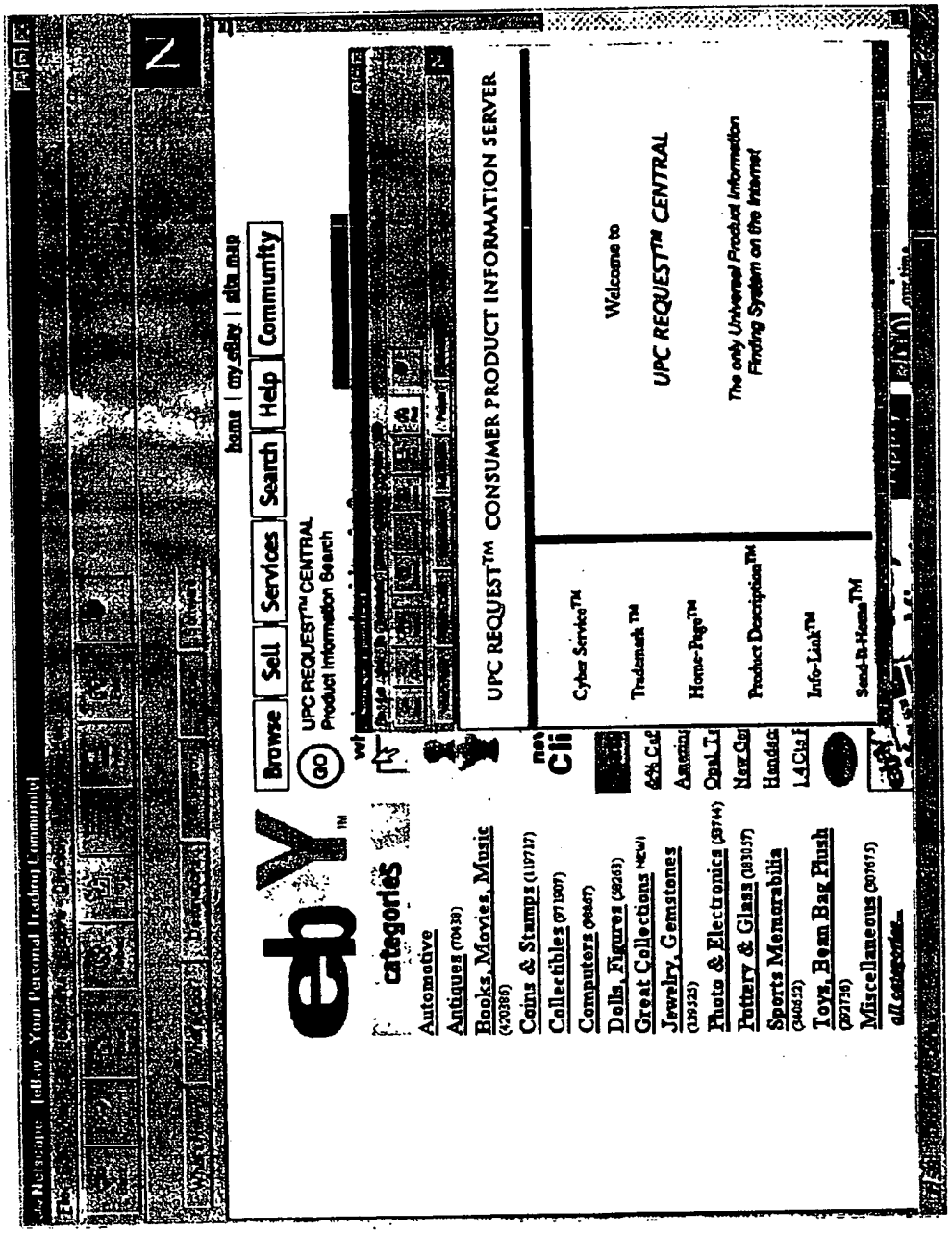

Browse
Sell
Services
Search
Help
Community
UPC REQUEST™ CENTRAL
Product Information Search
UPC REQUEST™ CONSUMER PRODUCT INFORMATION SERVER
Welcome to
UPC REQUEST™ CENTRAL
The only Universal Product Information Finding System on the Internet
Cyber Service™
Trademark ™
Home-Page™
Product Description™
Info-Link™
Send-It-Home™

FIG. 4Q2

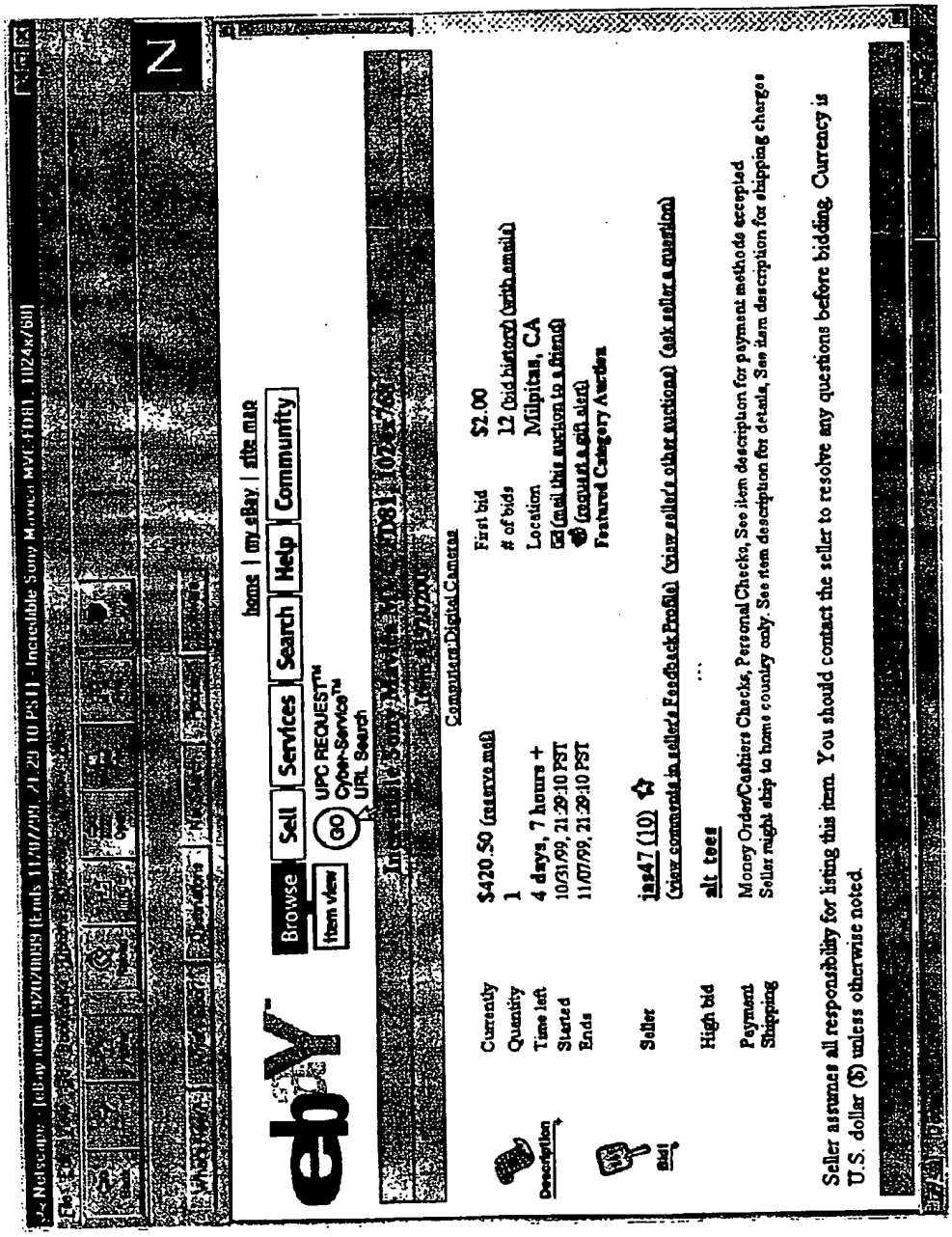

home | my eBay | site map
Browse
Sell
Services
Search
Help
Community
Item view
GO
UPC REQUEST™
Cyber-Service™
URL Search
Computers:Digital Cameras
Currently
Quantity
Time left
Started
Ends
Seller
High bid
Payment
Shipping
$420.50 (reserve met)
1
4 days, 7 hours +
10/31/99, 21:29:10 PST
11/07/99, 21:29:10 PST
ias47 (10)
(view comments in seller's Feedback Profile) (view seller's other auctions) (ask seller a question)
alt tees
Money Order/Cashiers Checks, Personal Checks, See item description for payment methods accepted
Seller might ship to home country only. See item description for details, See item description for shipping charges
First bid
# of bids
Location
$2.00
12 (bid history) (with email)
Milpitas, CA
(mail this auction to a friend)
(request a gift alert)
Featured Category Auction
Seller assumes all responsibility for listing this item. You should contact the seller to resolve any questions before bidding. Currency is U.S. dollar ($) unless otherwise noted.

FIG. 4R1

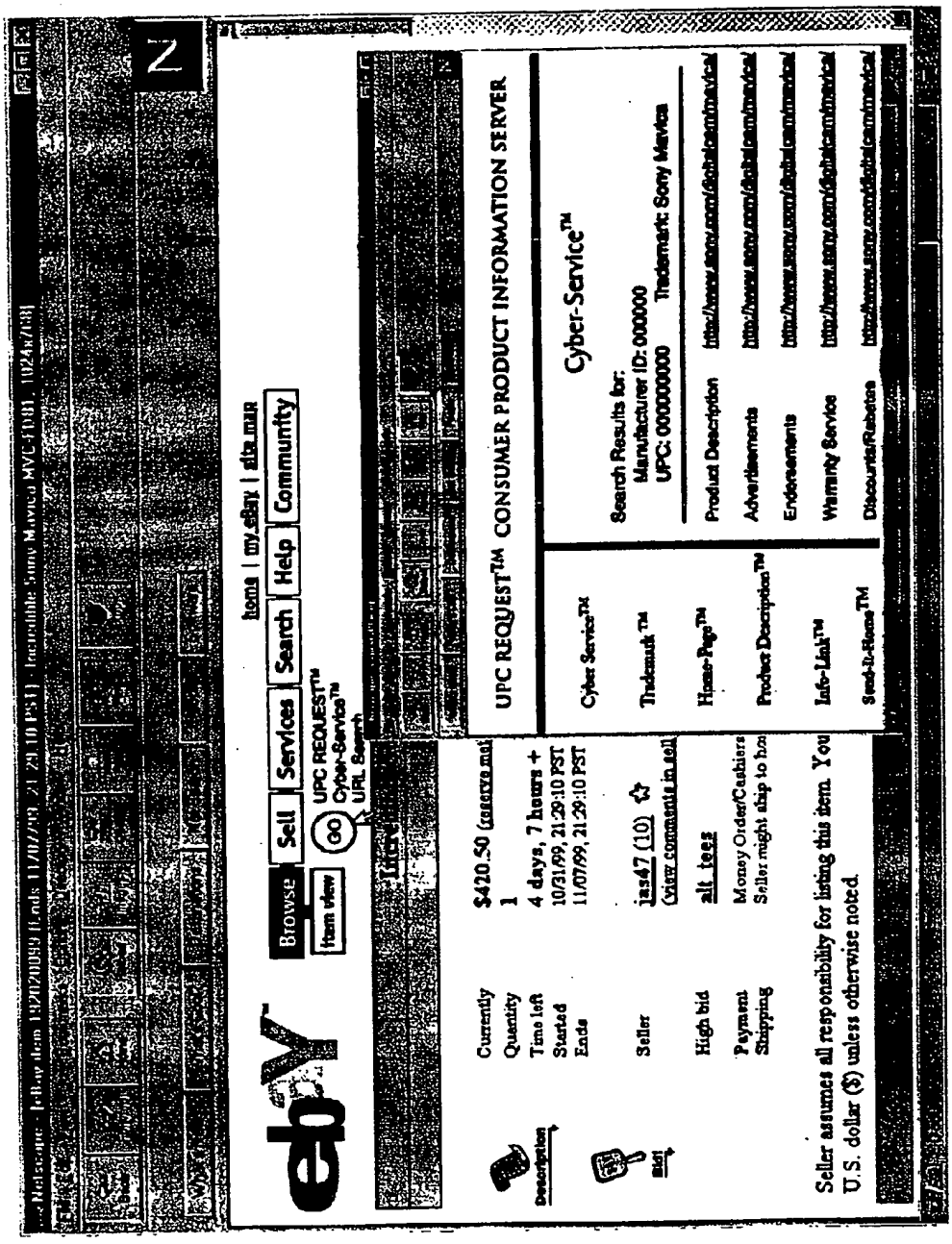

UPC REQUEST™ CONSUMER PRODUCT INFORMATION SERVER
Cyber-Service™
Search Results for:
Manufacturer ID: 000000
UPC: 000000000
Trademark: Sony Movies
Product Description
Advertisements
Endorsements
Warranty Service
Discounts/Rebates
Cyber Service™
Trademark™
Home-Page™
Product Description™
Info-Link™
Send-To-Friend™
Browse
Sell
Services
Search
Help
Community
UPC REQUEST™
Cyber-Service™
URL Search
Currently
Quantity
Time left
Started
Ends
Seller
High bid
Payment
Shipping
$420.50
1
4 days, 7 hours +
10/31/99, 21:29:10 PST
11/07/99, 21:29:10 PST
alt tags
Money Order/Cashiers
Seller assumes all responsibility for listing this item. You
U.S. dollar ($) unless otherwise noted.

FIG. 4R2

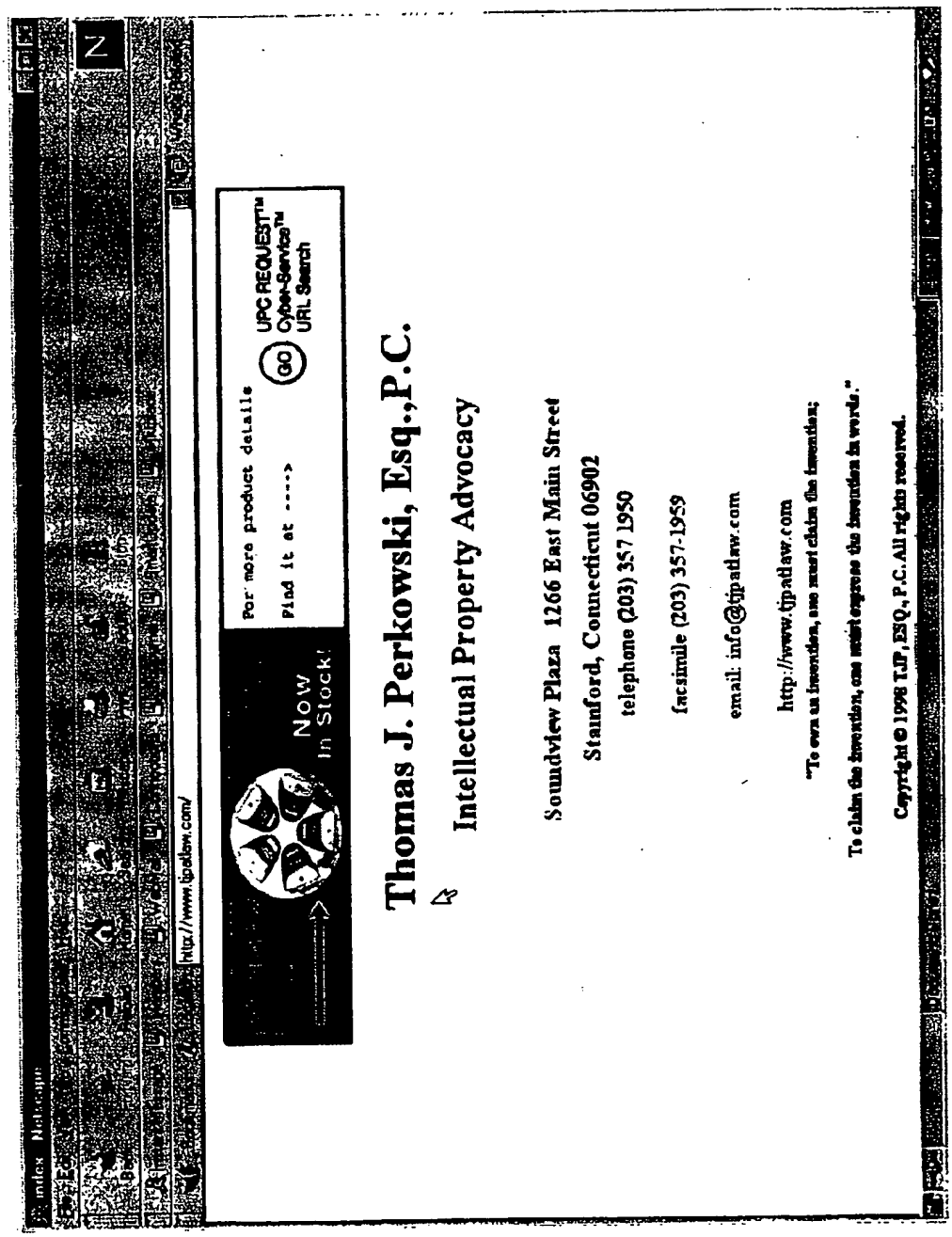

Thomas J. Perkowski, Esq.,P.C.
Intellectual Property Advocacy
Soundview Plaza 1266 East Main Street
Stamford, Connecticut 06902
telephone (203) 357 1950
facsimile (203) 357-1959
email: info@tjpatlaw.com
http://www.tjpatlaw.com
"To own an invention, one must claim the invention;
To claim the invention, one must express the invention in words."
Copyright © 1998 TJP, ESQ., P.C. All rights reserved.
Now In Stock!
For more product details
Find it at ---->
UPC REQUEST™
Cyber-Service™
URL Search

FIG. 4S1

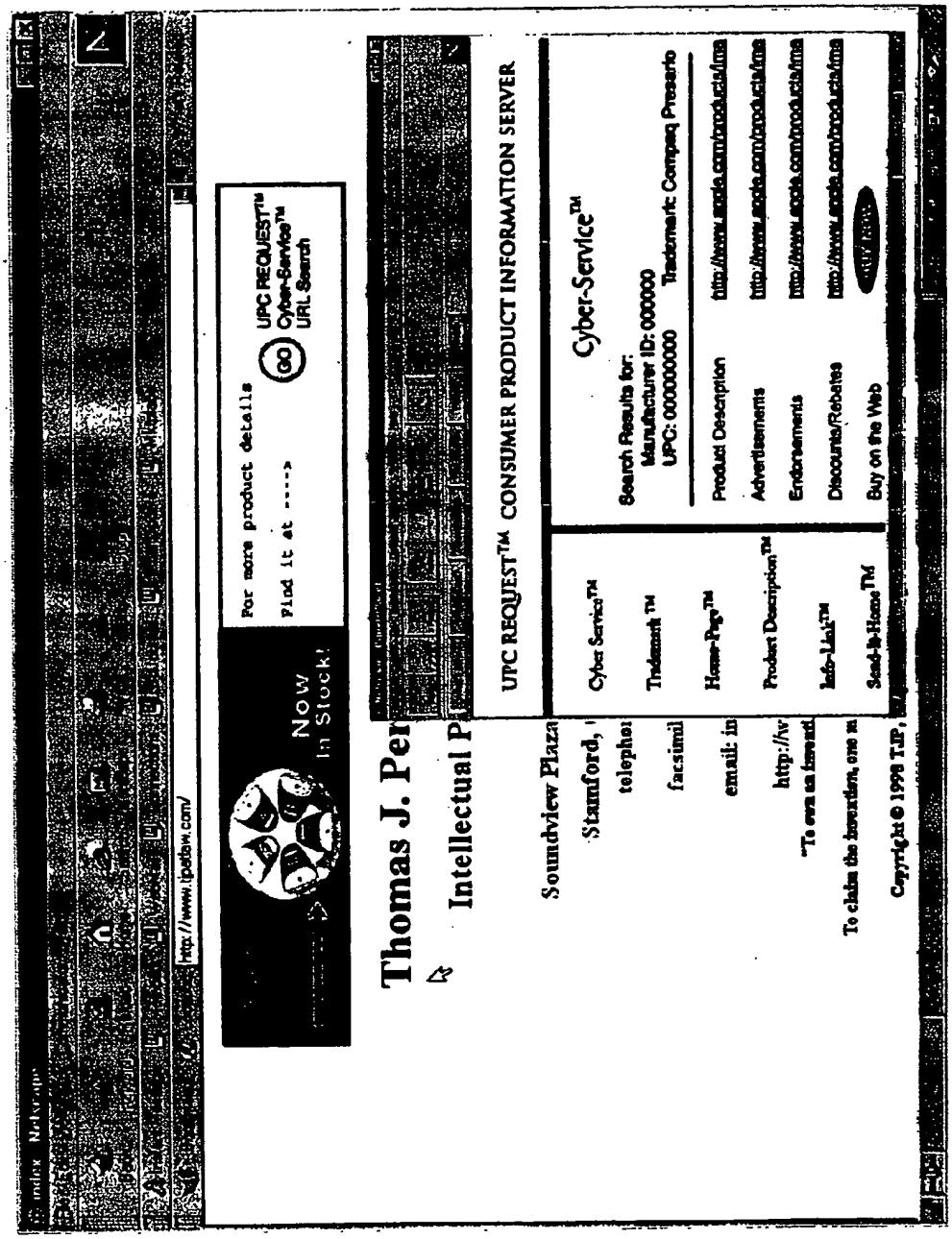

Thomas J. Per
Intellectual P
UPC REQUEST™ CONSUMER PRODUCT INFORMATION SERVER
Cyber-Service™
Search Results for:
Manufacturer ID: 000000
UPC: 000000000
Trademark: Compaq Presario
Product Description
Advertisements
Endorsements
Discounts/Rebates
Buy on the Web
Cyber Service™
Trademark™
Home-Page™
Product Description™
Info-Link™
Send-It-Home™
Soundview Plaza
Stamford,
telephon
facsimil
email: in
http://w
To claim the invention, one m
Copyright © 1998 T.P.
For more product details
Find it at ----->
UPC REQUEST™
Cyber-Service™
URL Search
Now
In Stock!
http://www.ipdow.com/

FIG. 4S2

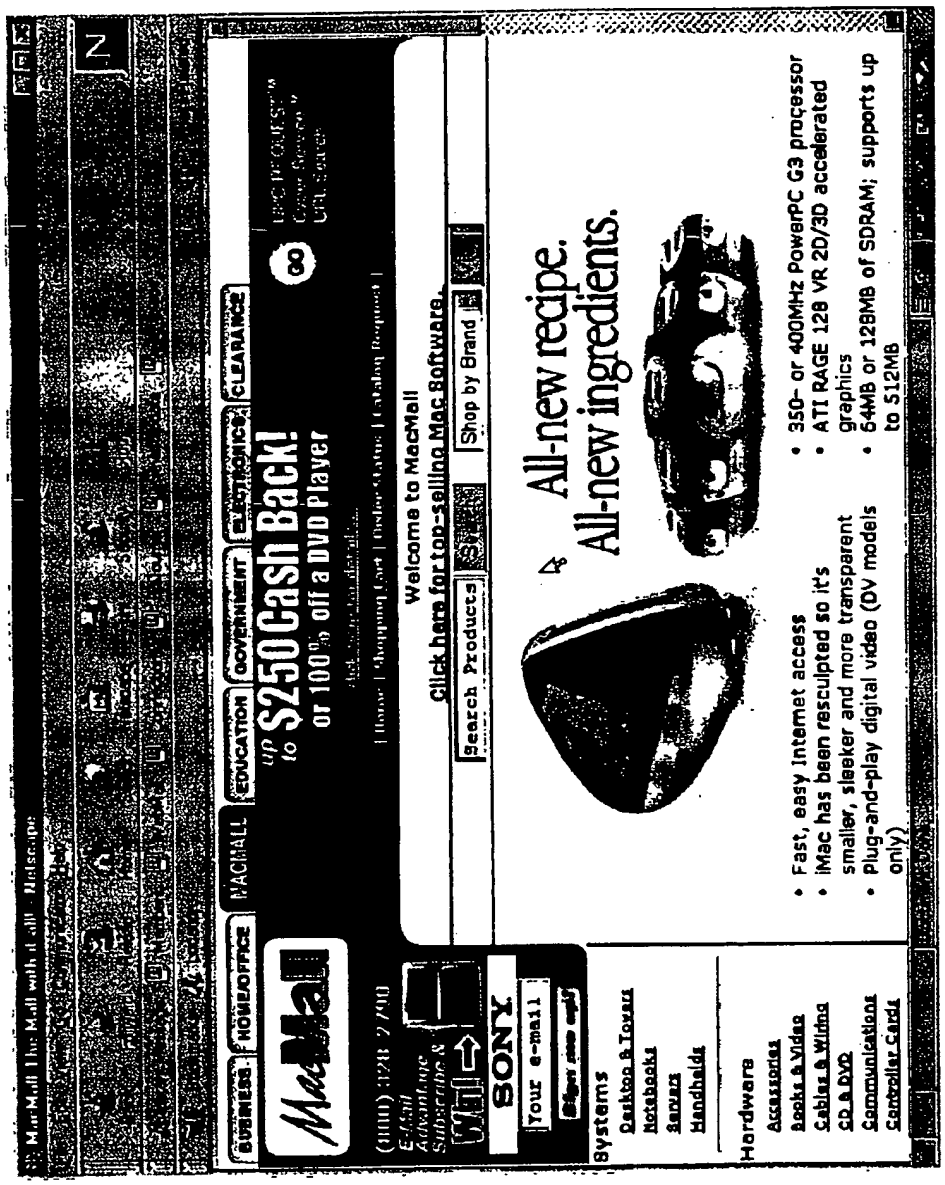

MacMall
BUSINESS
HOME/OFFICE
MACMALL
EDUCATION
GOVERNMENT
ELECTRONICS
CLEARANCE
up to $250 Cash Back!
or 100% off a DVD Player
Welcome to MacMall
Click here for top-selling Mac Software
Search Products
Shop by Brand
SONY
Your e-mail
Systems
Desktop & Towers
Notebooks
Servers
Handhelds
Hardware
Accessories
Books & Video
Cables & Wires
CD & DVD
Communications
Controller Cards
All-new recipe.
All-new ingredients.
• Fast, easy Internet access
• iMac has been resculpted so it's smaller, sleeker and more transparent
• Plug-and-play digital video (DV models only)
• 350- or 400MHz PowerPC G3 processor
• ATI RAGE 128 VR 2D/3D accelerated graphics
• 64MB or 128MB of SDRAM; supports up to 512MB

FIG. 4S3

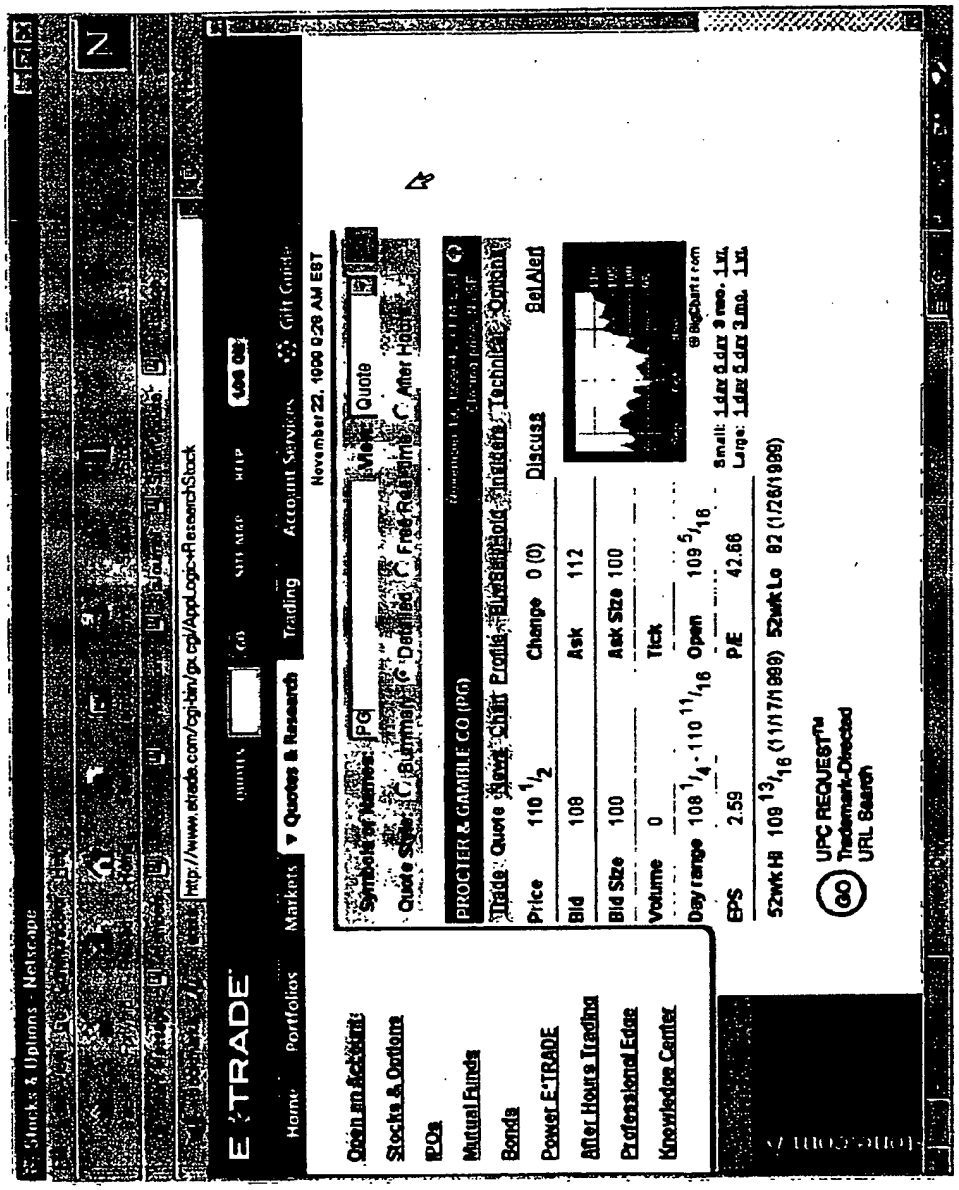

Stocks & Options - Netscape
E*TRADE
Home
Portfolios
Markets
Quotes & Research
Trading
Account Services
Gift Guide
November 22, 1999 9:28 AM EST
Symbol or Names: PG
Quote
Quote Style: Summary Detailed Free Real-time After Hours
PROCTER & GAMBLE CO (PG)
Trade Quote Chart Profile Fundamentals Technicals Options
Price 110 1/2
Change 0 (0)
Discuss
Bid Alert
Bid 108
Ask 112
Bid Size 100
Ask Size 100
Volume 0
Tick
Day range 108 1/4 - 110 11/16
Open 109 5/16
EPS 2.59
P/E 42.66
52wk Hi 109 13/16 (11/17/1999)
52wk Lo 82 (1/28/1999)
UPC REQUEST™
Trademark-Directed
URL Search
Options Center
Order Entry
Stocks & Options
IPOs
Mutual Funds
Bonds
Power E*TRADE
After Hours Trading
Professional Edge
Knowledge Center

FIG. 4T1

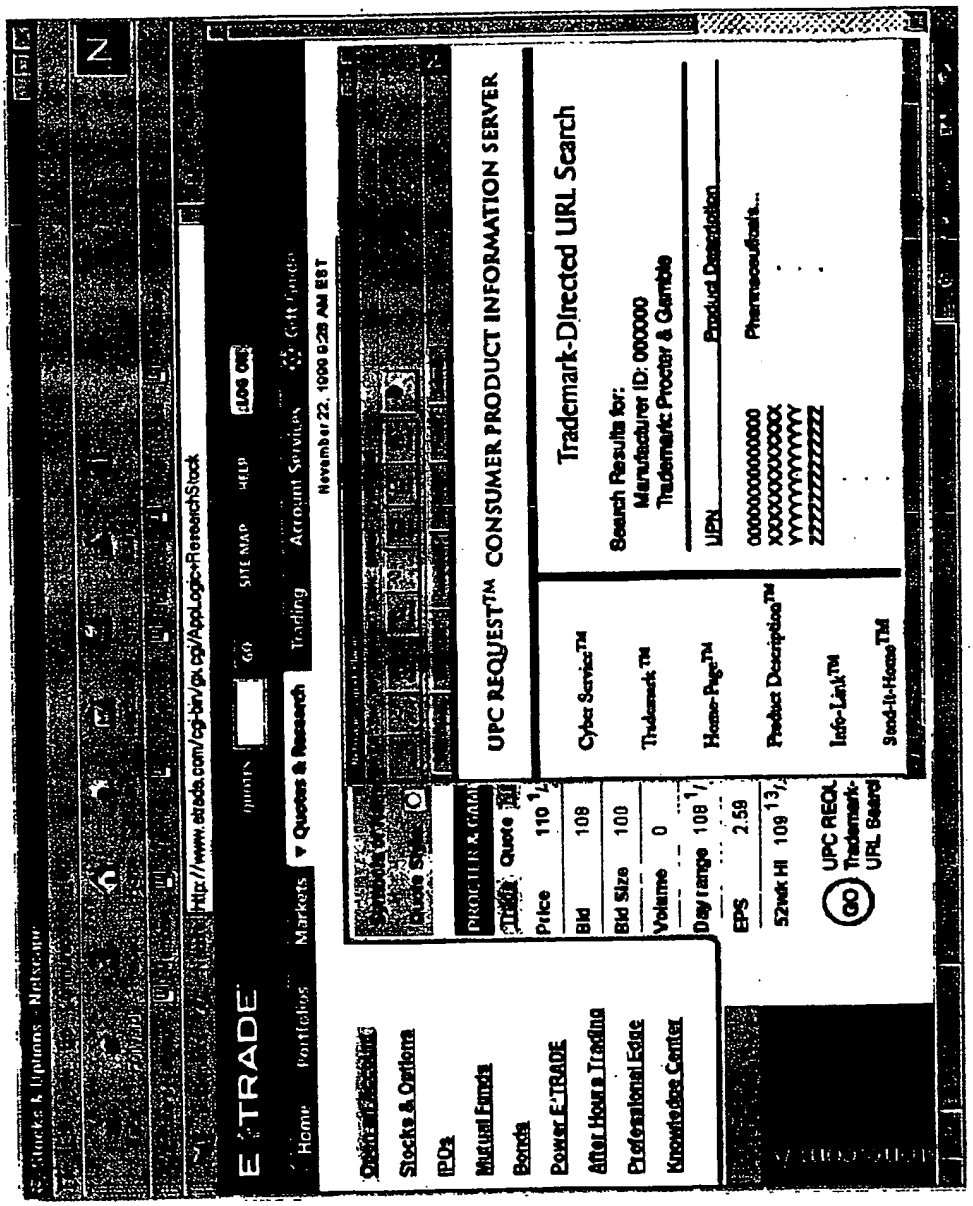

E*TRADE
Home
Portfolios
Markets
Quotes & Research
Trading
Account Services
Gift Certificates
SITE MAP
HELP
LOG OFF
November 22, 1999 9:28 AM EST
Stocks & Options
IPOs
Mutual Funds
Bonds
Power E*TRADE
After Hours Trading
Professional Edge
Knowledge Center
UPC REQUEST™ CONSUMER PRODUCT INFORMATION SERVER
Trademark-Directed URL Search
Search Results for:
Manufacturer ID: 000000
Trademark: Procter & Gamble
UPN
Product Description
000000000000 Pharmaceuticals...
XXXXXXXXXXXX
YYYYYYYYYYYY
ZZZZZZZZZZZZ
Price 110 1/2
Bid 108
Bid Size 100
Volume 0
Day range 108 1/
EPS 2.59
52wk HI 109 13/
Cyber Service™
Trademark ™
Home-Page™
Product Description™
Info-Link™
Send-It-Home™
GO
UPC REQU
Trademark-
URL Searc

FIG. 4T2

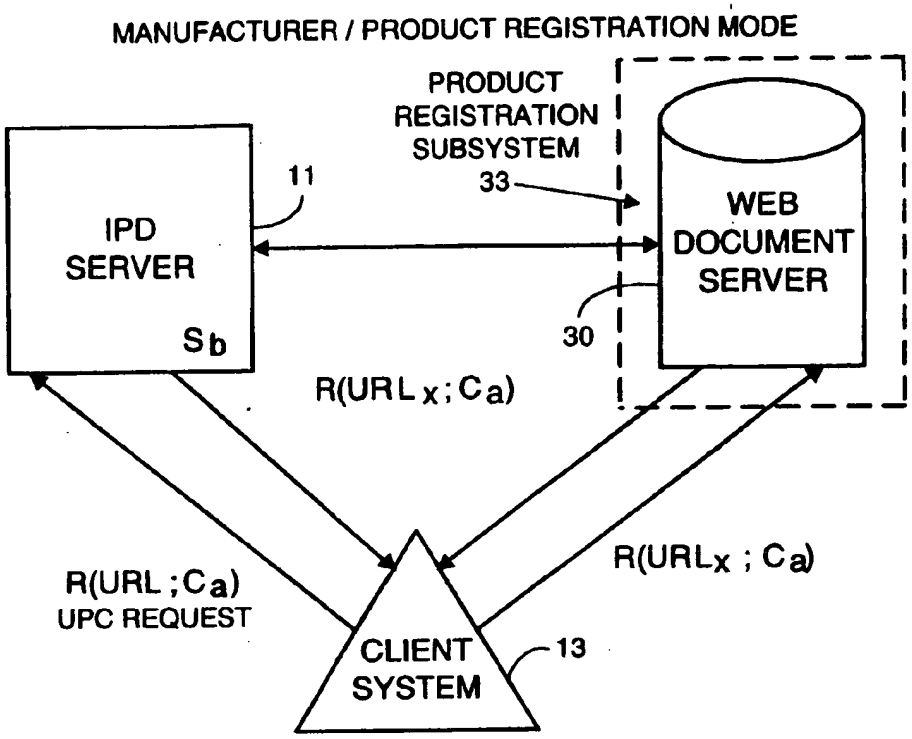

MANUFACTURER / PRODUCT REGISTRATION MODE
PRODUCT REGISTRATION SUBSYSTEM
33
11
IPD SERVER
Sb
WEB DOCUMENT SERVER
30
R(URLx;Ca)
R(URL;Ca)
UPC REQUEST
R(URLx ; Ca)
CLIENT SYSTEM
13

FIG. 5A

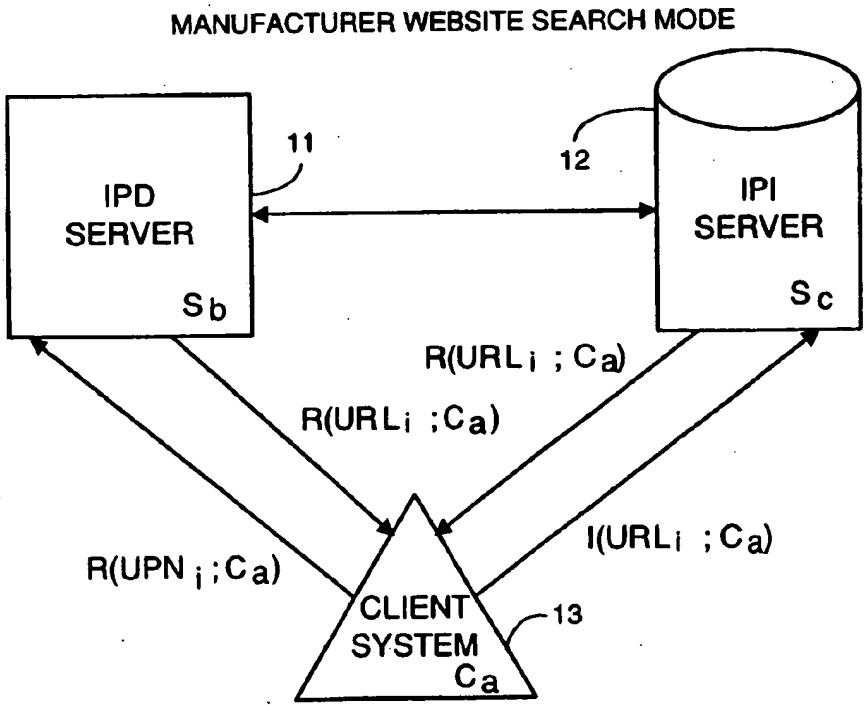

MANUFACTURER WEBSITE SEARCH MODE
11
12
IPD SERVER
Sb
IPI SERVER
Sc
R(URLi ; Ca)
R(URLi ;Ca)
R(UPNi;Ca)
I(URLi ; Ca)
CLIENT SYSTEM
Ca
13

FIG. 5B

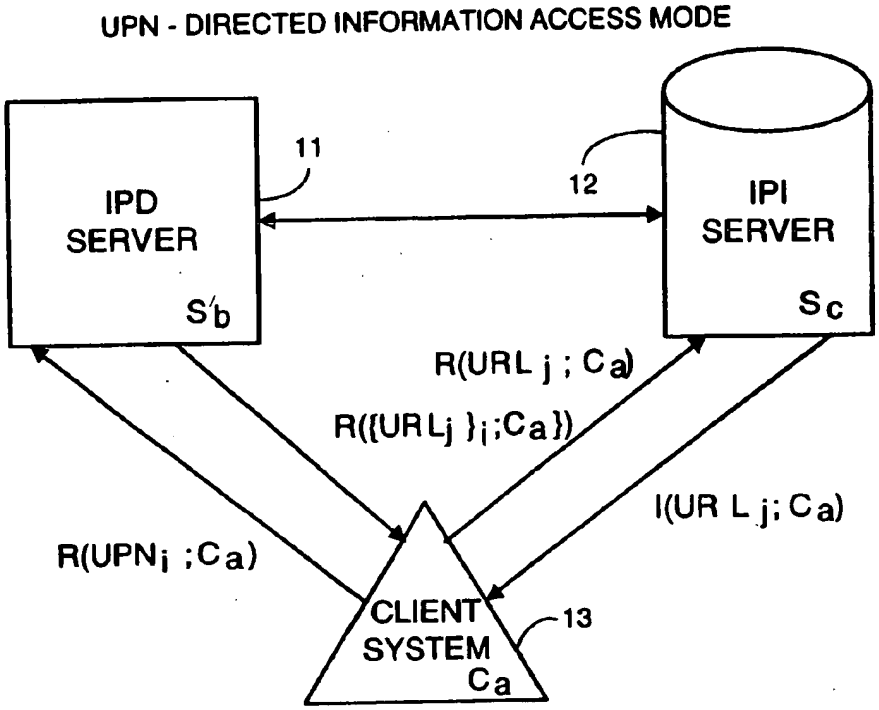

UPN - DIRECTED INFORMATION ACCESS MODE
IPD SERVER
S'b
11
12
IPI SERVER
Sc
R(URL j ; Ca)
R({URLj }i ;Ca})
I(UR L j; Ca)
R(UPNi ;Ca)
CLIENT SYSTEM
Ca
13

FIG. 5C

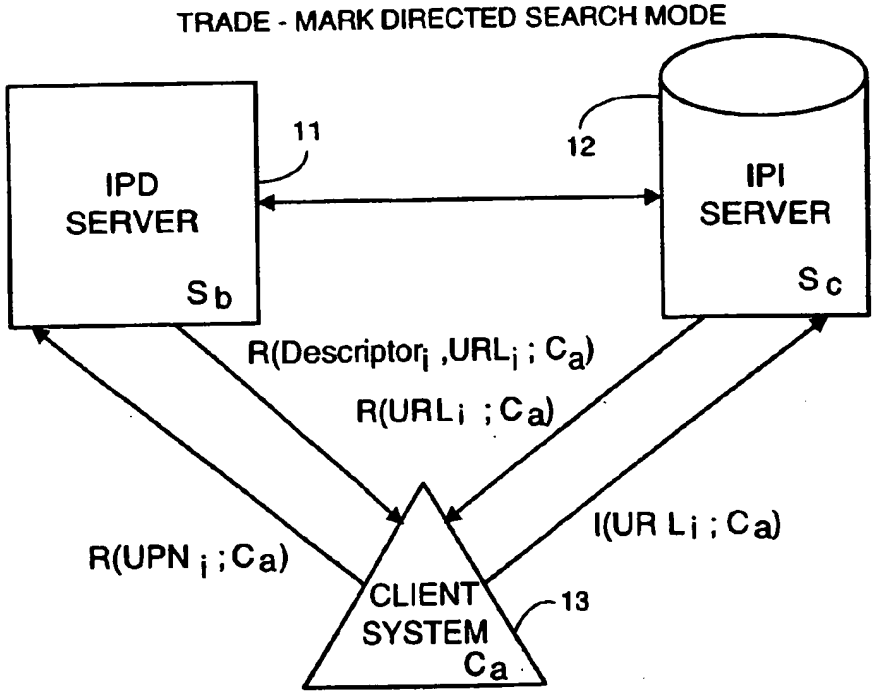

TRADE - MARK DIRECTED SEARCH MODE
IPD SERVER
Sb
11
12
IPI SERVER
Sc
R(Descriptori ,URLi ; Ca)
R(URLi ; Ca)
I(UR Li ; Ca)
R(UPN i ;Ca)
CLIENT SYSTEM
Ca
13

FIG. 5D

PRODUCT - DESCRIPTION DIRECTED SEARCH MODE

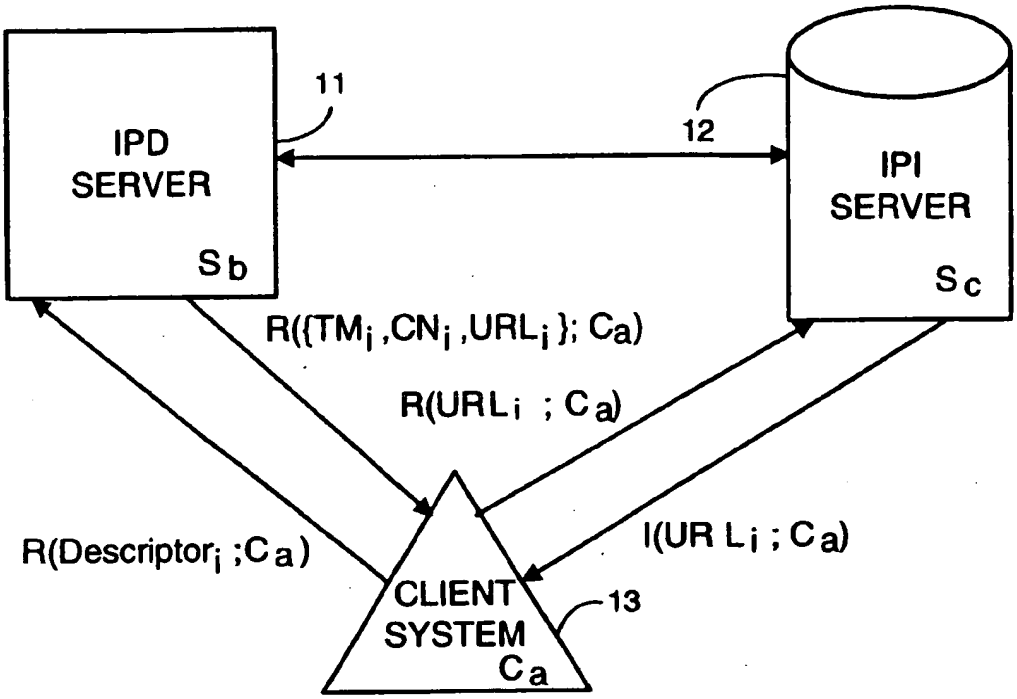


FIG. 5E

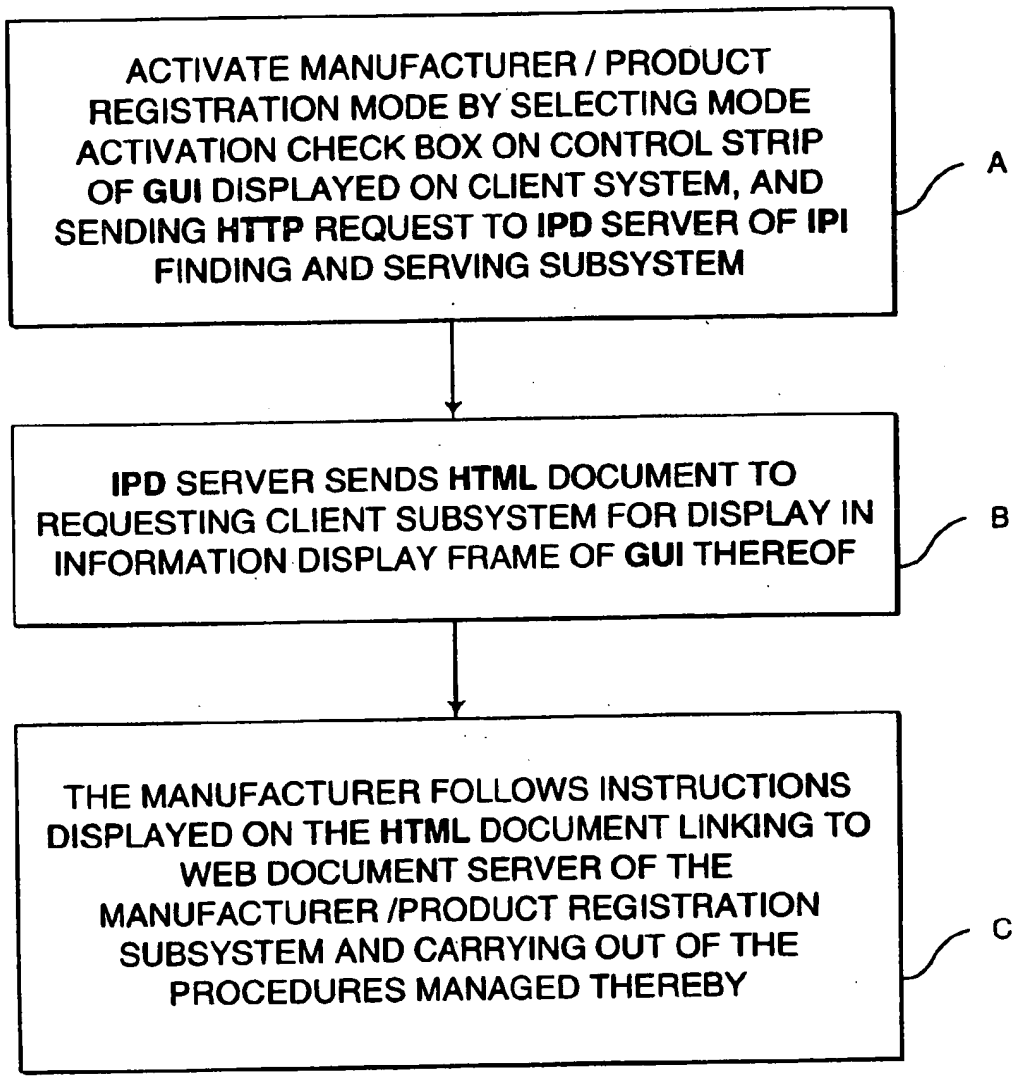

ACTIVATE MANUFACTURER / PRODUCT REGISTRATION MODE BY SELECTING MODE ACTIVATION CHECK BOX ON CONTROL STRIP OF GUI DISPLAYED ON CLIENT SYSTEM, AND SENDING HTTP REQUEST TO IPD SERVER OF IPI FINDING AND SERVING SUBSYSTEM
A
IPD SERVER SENDS HTML DOCUMENT TO REQUESTING CLIENT SUBSYSTEM FOR DISPLAY IN INFORMATION DISPLAY FRAME OF GUI THEREOF
B
THE MANUFACTURER FOLLOWS INSTRUCTIONS DISPLAYED ON THE HTML DOCUMENT LINKING TO WEB DOCUMENT SERVER OF THE MANUFACTURER /PRODUCT REGISTRATION SUBSYSTEM AND CARRYING OUT OF THE PROCEDURES MANAGED THEREBY
C

FIG. 6A

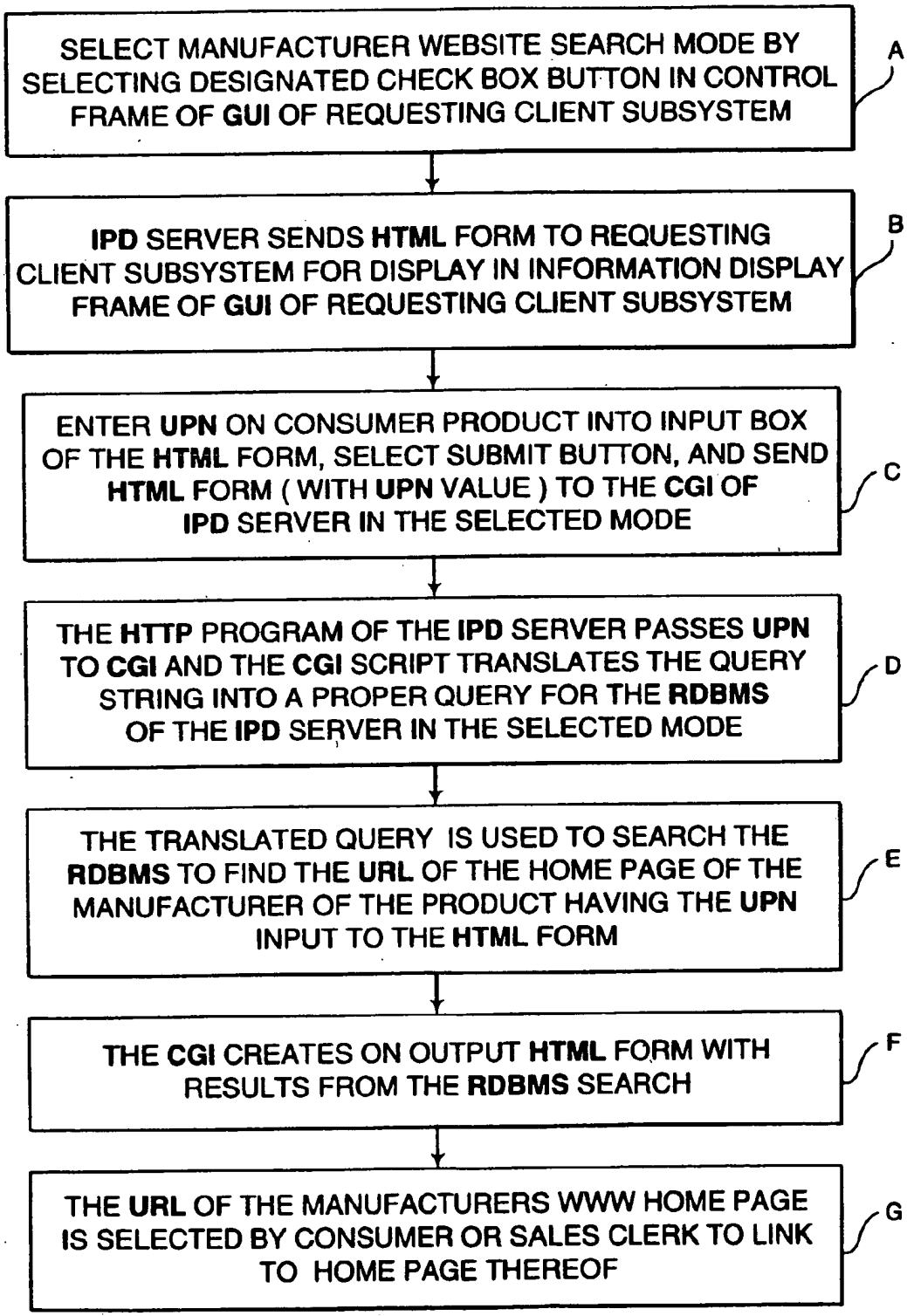

SELECT MANUFACTURER WEBSITE SEARCH MODE BY SELECTING DESIGNATED CHECK BOX BUTTON IN CONTROL FRAME OF GUI OF REQUESTING CLIENT SUBSYSTEM
A
IPD SERVER SENDS HTML FORM TO REQUESTING CLIENT SUBSYSTEM FOR DISPLAY IN INFORMATION DISPLAY FRAME OF GUI OF REQUESTING CLIENT SUBSYSTEM
B
ENTER UPN ON CONSUMER PRODUCT INTO INPUT BOX OF THE HTML FORM, SELECT SUBMIT BUTTON, AND SEND HTML FORM (WITH UPN VALUE) TO THE CGI OF IPD SERVER IN THE SELECTED MODE
C
THE HTTP PROGRAM OF THE IPD SERVER PASSES UPN TO CGI AND THE CGI SCRIPT TRANSLATES THE QUERY STRING INTO A PROPER QUERY FOR THE RDBMS OF THE IPD SERVER IN THE SELECTED MODE
D
THE TRANSLATED QUERY IS USED TO SEARCH THE RDBMS TO FIND THE URL OF THE HOME PAGE OF THE MANUFACTURER OF THE PRODUCT HAVING THE UPN INPUT TO THE HTML FORM
E
THE CGI CREATES ON OUTPUT HTML FORM WITH RESULTS FROM THE RDBMS SEARCH
F
THE URL OF THE MANUFACTURERS WWW HOME PAGE IS SELECTED BY CONSUMER OR SALES CLERK TO LINK TO HOME PAGE THEREOF
G

FIG. 6B

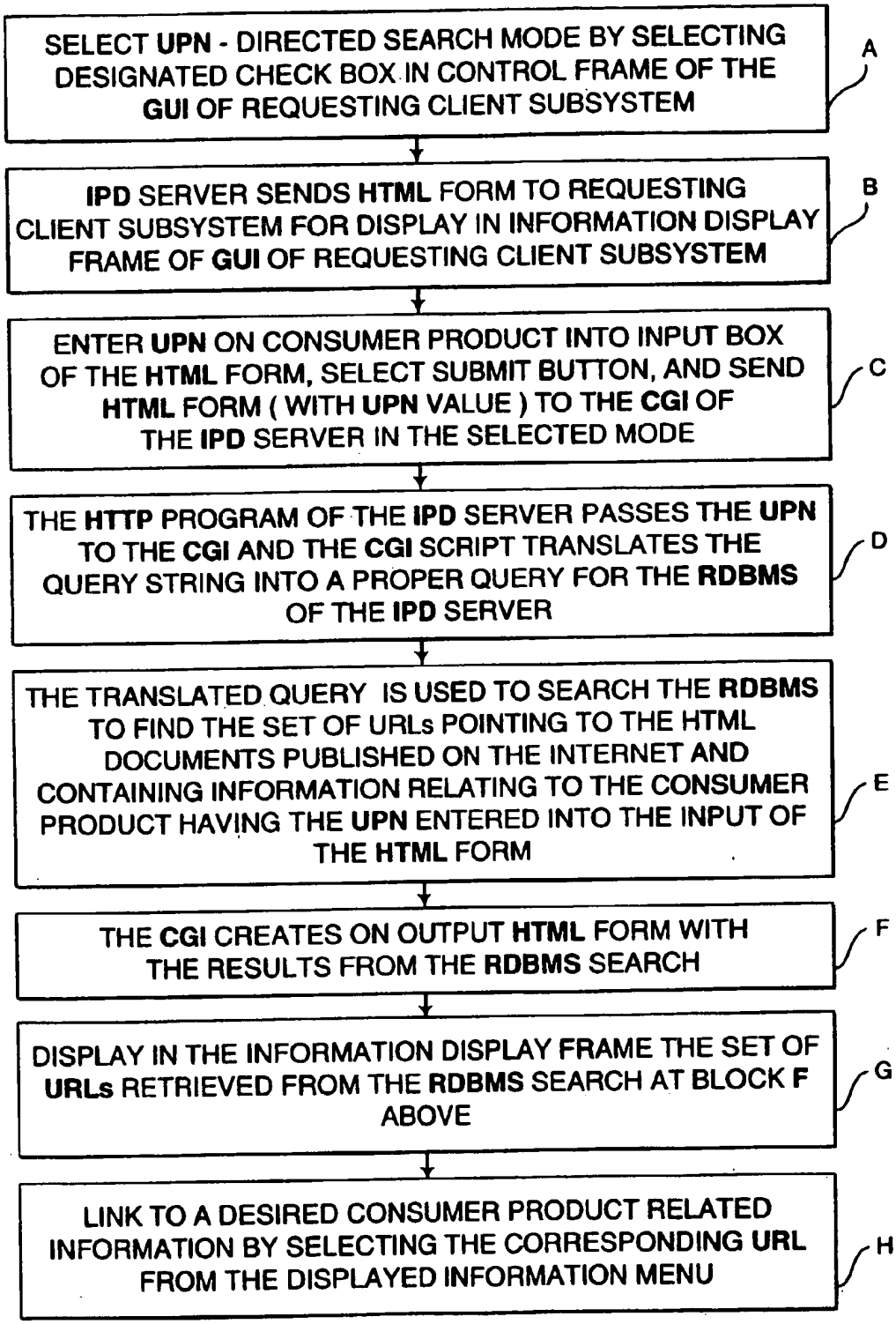

SELECT UPN - DIRECTED SEARCH MODE BY SELECTING DESIGNATED CHECK BOX IN CONTROL FRAME OF THE GUI OF REQUESTING CLIENT SUBSYSTEM
A
IPD SERVER SENDS HTML FORM TO REQUESTING CLIENT SUBSYSTEM FOR DISPLAY IN INFORMATION DISPLAY FRAME OF GUI OF REQUESTING CLIENT SUBSYSTEM
B
ENTER UPN ON CONSUMER PRODUCT INTO INPUT BOX OF THE HTML FORM, SELECT SUBMIT BUTTON, AND SEND HTML FORM (WITH UPN VALUE) TO THE CGI OF THE IPD SERVER IN THE SELECTED MODE
C
THE HTTP PROGRAM OF THE IPD SERVER PASSES THE UPN TO THE CGI AND THE CGI SCRIPT TRANSLATES THE QUERY STRING INTO A PROPER QUERY FOR THE RDBMS OF THE IPD SERVER
D
THE TRANSLATED QUERY IS USED TO SEARCH THE RDBMS TO FIND THE SET OF URLs POINTING TO THE HTML DOCUMENTS PUBLISHED ON THE INTERNET AND CONTAINING INFORMATION RELATING TO THE CONSUMER PRODUCT HAVING THE UPN ENTERED INTO THE INPUT OF THE HTML FORM
E
THE CGI CREATES ON OUTPUT HTML FORM WITH THE RESULTS FROM THE RDBMS SEARCH
F
DISPLAY IN THE INFORMATION DISPLAY FRAME THE SET OF URLs RETRIEVED FROM THE RDBMS SEARCH AT BLOCK F ABOVE
G
LINK TO A DESIRED CONSUMER PRODUCT RELATED INFORMATION BY SELECTING THE CORRESPONDING URL FROM THE DISPLAYED INFORMATION MENU
H

FIG. 6C

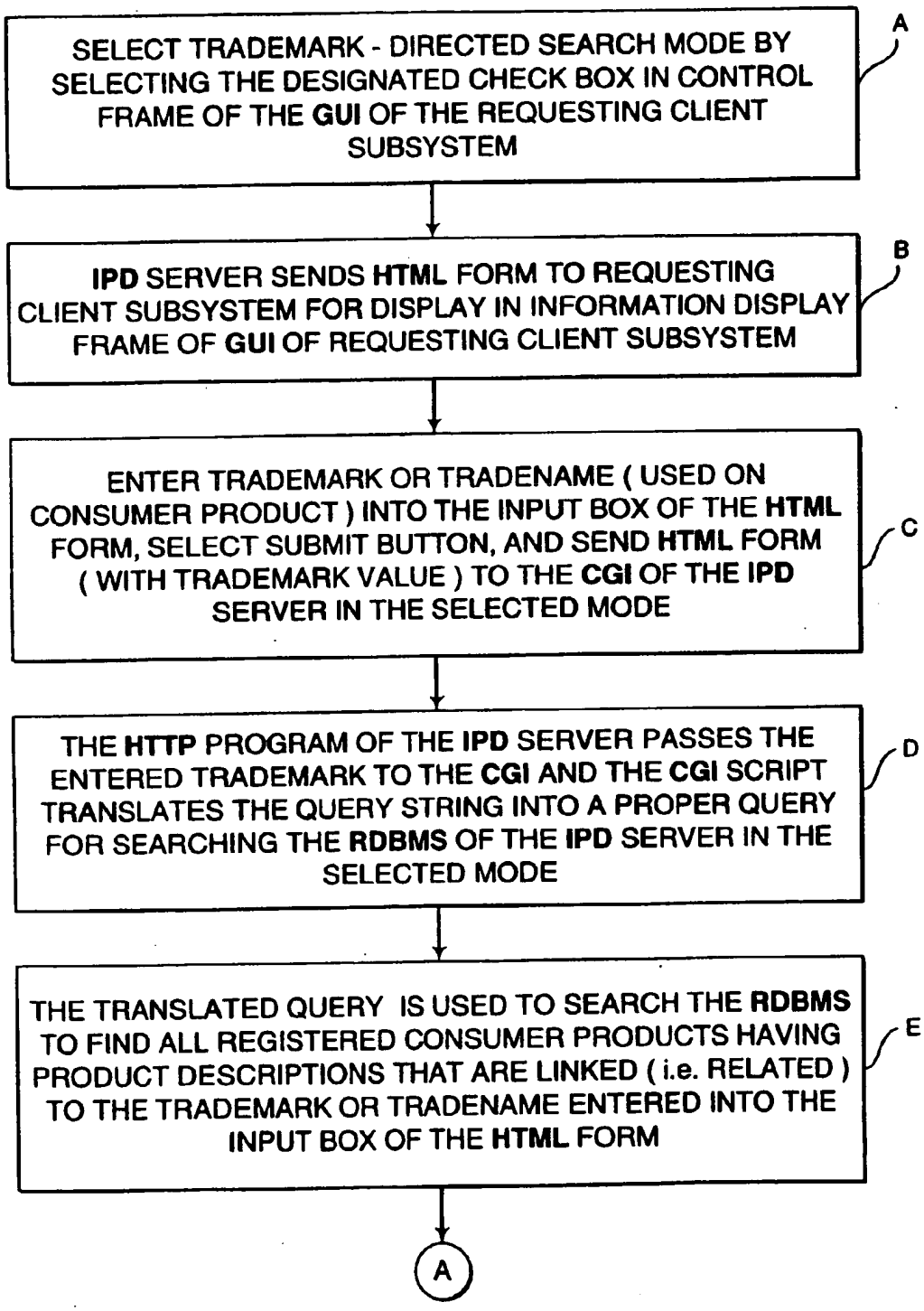
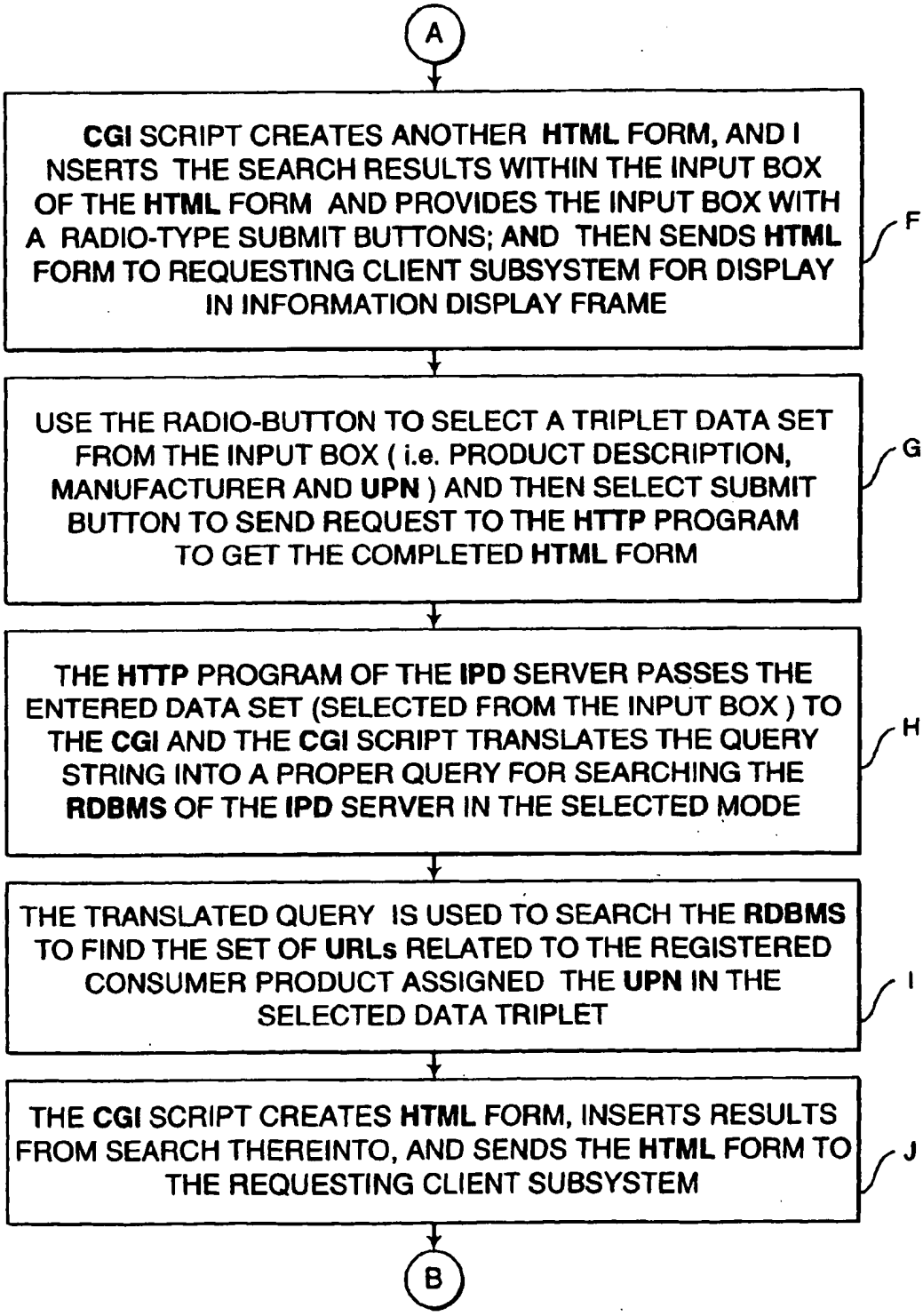

SELECT TRADEMARK - DIRECTED SEARCH MODE BY SELECTING THE DESIGNATED CHECK BOX IN CONTROL FRAME OF THE GUI OF THE REQUESTING CLIENT SUBSYSTEM
A
IPD SERVER SENDS HTML FORM TO REQUESTING CLIENT SUBSYSTEM FOR DISPLAY IN INFORMATION DISPLAY FRAME OF GUI OF REQUESTING CLIENT SUBSYSTEM
B
ENTER TRADEMARK OR TRADENAME (USED ON CONSUMER PRODUCT) INTO THE INPUT BOX OF THE HTML FORM, SELECT SUBMIT BUTTON, AND SEND HTML FORM (WITH TRADEMARK VALUE) TO THE CGI OF THE IPD SERVER IN THE SELECTED MODE
C
THE HTTP PROGRAM OF THE IPD SERVER PASSES THE ENTERED TRADEMARK TO THE CGI AND THE CGI SCRIPT TRANSLATES THE QUERY STRING INTO A PROPER QUERY FOR SEARCHING THE RDBMS OF THE IPD SERVER IN THE SELECTED MODE
D
THE TRANSLATED QUERY IS USED TO SEARCH THE RDBMS TO FIND ALL REGISTERED CONSUMER PRODUCTS HAVING PRODUCT DESCRIPTIONS THAT ARE LINKED (i.e. RELATED) TO THE TRADEMARK OR TRADENAME ENTERED INTO THE INPUT BOX OF THE HTML FORM
E
A

FIG. 6D1


A
CGI SCRIPT CREATES ANOTHER HTML FORM, AND I NSERTS THE SEARCH RESULTS WITHIN THE INPUT BOX OF THE HTML FORM AND PROVIDES THE INPUT BOX WITH A RADIO-TYPE SUBMIT BUTTONS; AND THEN SENDS HTML FORM TO REQUESTING CLIENT SUBSYSTEM FOR DISPLAY IN INFORMATION DISPLAY FRAME
F
USE THE RADIO-BUTTON TO SELECT A TRIPLET DATA SET FROM THE INPUT BOX (i.e. PRODUCT DESCRIPTION, MANUFACTURER AND UPN) AND THEN SELECT SUBMIT BUTTON TO SEND REQUEST TO THE HTTP PROGRAM TO GET THE COMPLETED HTML FORM
G
THE HTTP PROGRAM OF THE IPD SERVER PASSES THE ENTERED DATA SET (SELECTED FROM THE INPUT BOX) TO THE CGI AND THE CGI SCRIPT TRANSLATES THE QUERY STRING INTO A PROPER QUERY FOR SEARCHING THE RDBMS OF THE IPD SERVER IN THE SELECTED MODE
H
THE TRANSLATED QUERY IS USED TO SEARCH THE RDBMS TO FIND THE SET OF URLs RELATED TO THE REGISTERED CONSUMER PRODUCT ASSIGNED THE UPN IN THE SELECTED DATA TRIPLET
I
THE CGI SCRIPT CREATES HTML FORM, INSERTS RESULTS FROM SEARCH THEREINTO, AND SENDS THE HTML FORM TO THE REQUESTING CLIENT SUBSYSTEM
J
B

F I G. 6D2

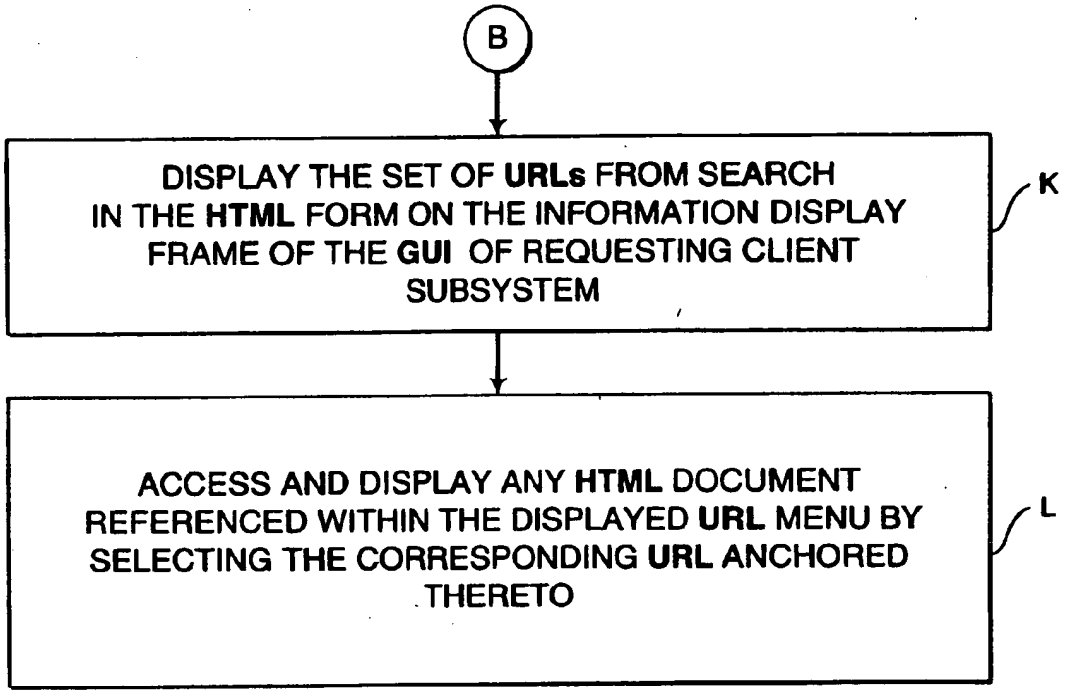

B
DISPLAY THE SET OF URLs FROM SEARCH IN THE HTML FORM ON THE INFORMATION DISPLAY FRAME OF THE GUI OF REQUESTING CLIENT SUBSYSTEM
K
ACCESS AND DISPLAY ANY HTML DOCUMENT REFERENCED WITHIN THE DISPLAYED URL MENU BY SELECTING THE CORRESPONDING URL ANCHORED THERETO
L

FIG. 6D3

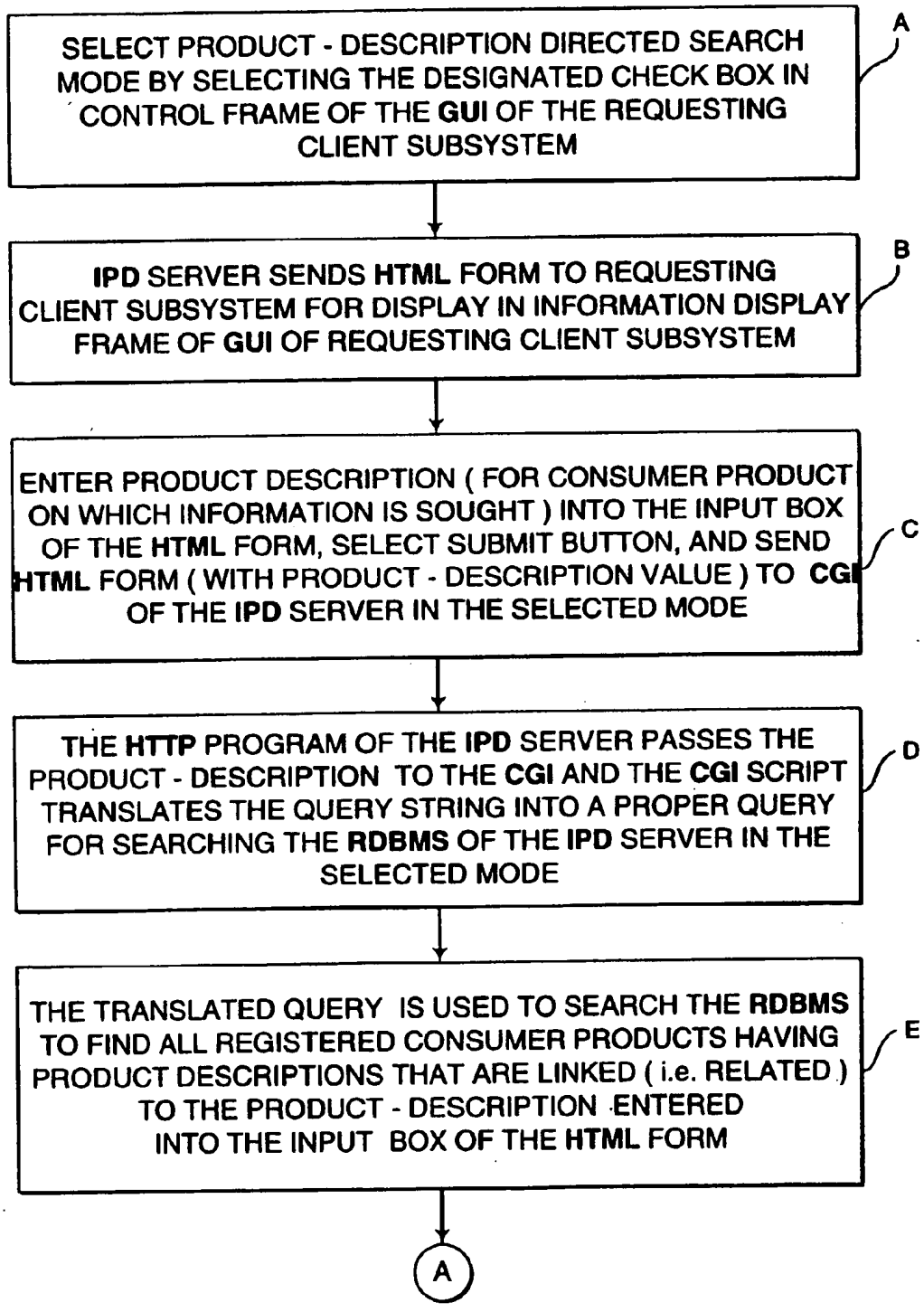

SELECT PRODUCT - DESCRIPTION DIRECTED SEARCH MODE BY SELECTING THE DESIGNATED CHECK BOX IN CONTROL FRAME OF THE GUI OF THE REQUESTING CLIENT SUBSYSTEM
A
IPD SERVER SENDS HTML FORM TO REQUESTING CLIENT SUBSYSTEM FOR DISPLAY IN INFORMATION DISPLAY FRAME OF GUI OF REQUESTING CLIENT SUBSYSTEM
B
ENTER PRODUCT DESCRIPTION (FOR CONSUMER PRODUCT ON WHICH INFORMATION IS SOUGHT) INTO THE INPUT BOX OF THE HTML FORM, SELECT SUBMIT BUTTON, AND SEND HTML FORM (WITH PRODUCT - DESCRIPTION VALUE) TO CGI OF THE IPD SERVER IN THE SELECTED MODE
C
THE HTTP PROGRAM OF THE IPD SERVER PASSES THE PRODUCT - DESCRIPTION TO THE CGI AND THE CGI SCRIPT TRANSLATES THE QUERY STRING INTO A PROPER QUERY FOR SEARCHING THE RDBMS OF THE IPD SERVER IN THE SELECTED MODE
D
THE TRANSLATED QUERY IS USED TO SEARCH THE RDBMS TO FIND ALL REGISTERED CONSUMER PRODUCTS HAVING PRODUCT DESCRIPTIONS THAT ARE LINKED (i.e. RELATED) TO THE PRODUCT - DESCRIPTION ENTERED INTO THE INPUT BOX OF THE HTML FORM
E
A

FIG. 6E1

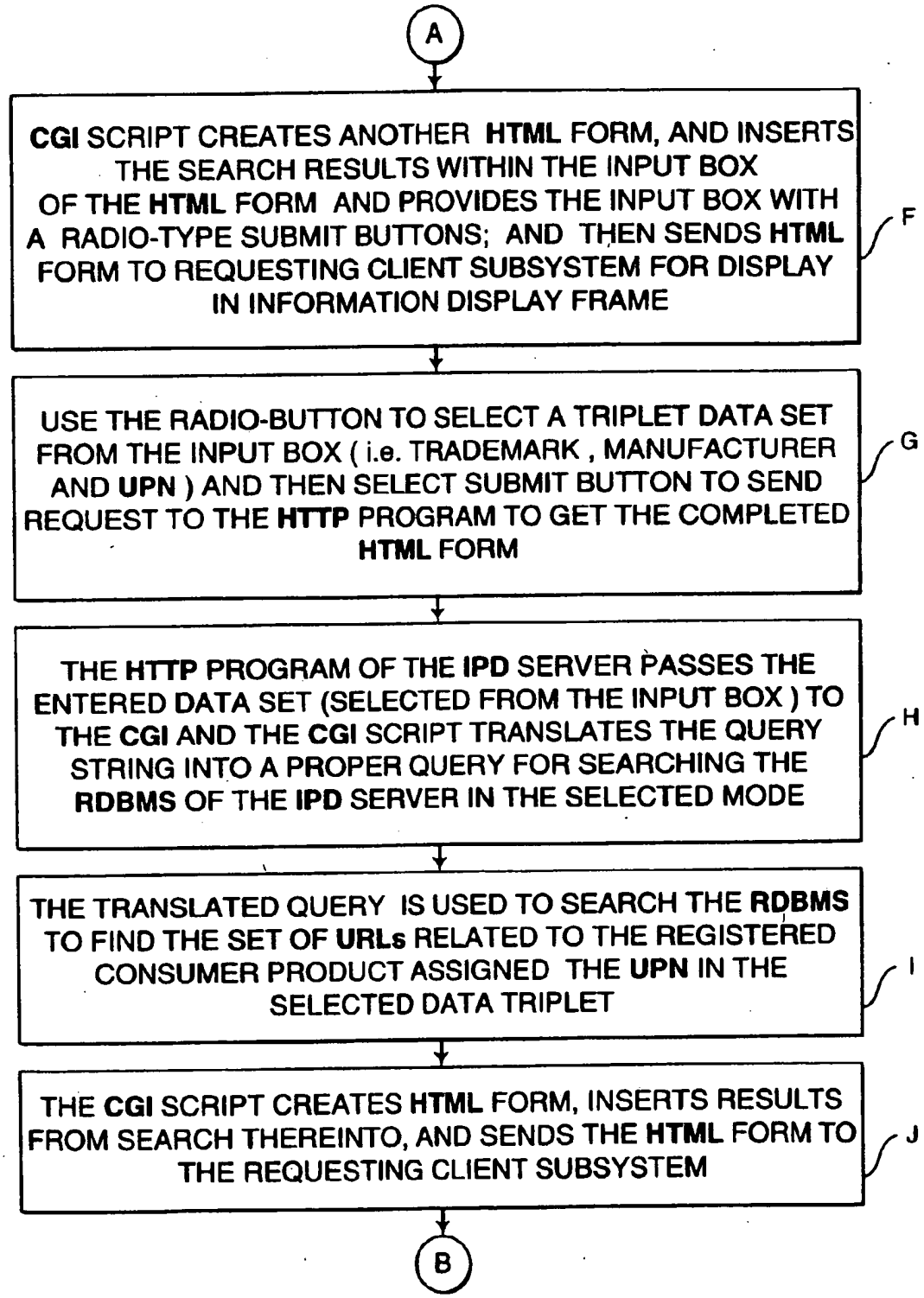
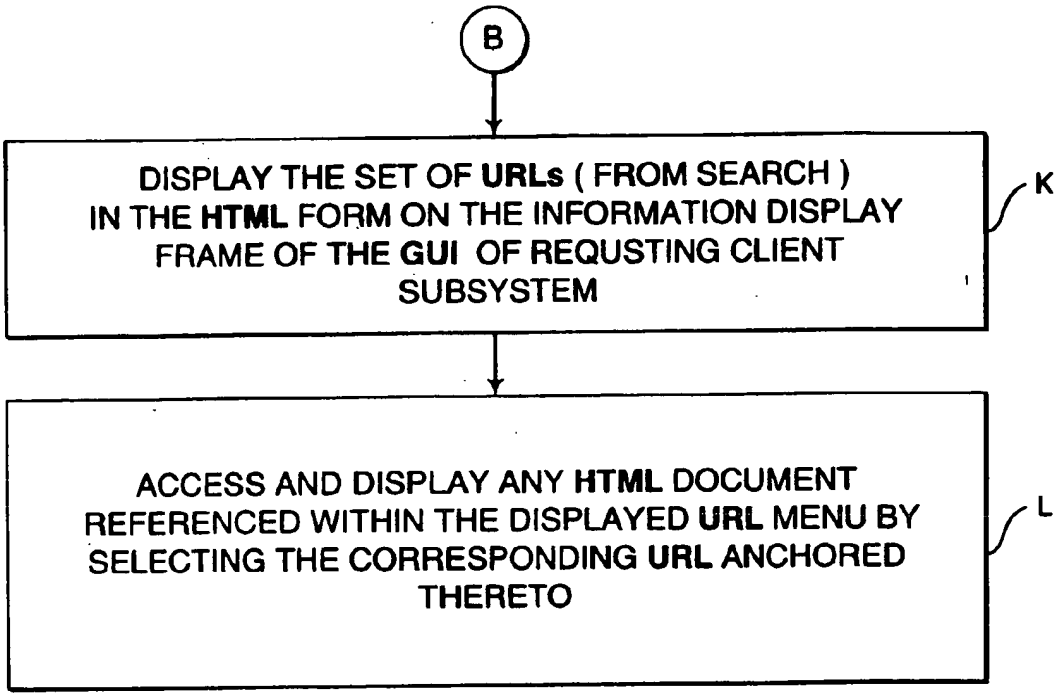

A
CGI SCRIPT CREATES ANOTHER HTML FORM, AND INSERTS THE SEARCH RESULTS WITHIN THE INPUT BOX OF THE HTML FORM AND PROVIDES THE INPUT BOX WITH A RADIO-TYPE SUBMIT BUTTONS; AND THEN SENDS HTML FORM TO REQUESTING CLIENT SUBSYSTEM FOR DISPLAY IN INFORMATION DISPLAY FRAME
F
USE THE RADIO-BUTTON TO SELECT A TRIPLET DATA SET FROM THE INPUT BOX (i.e. TRADEMARK , MANUFACTURER AND UPN) AND THEN SELECT SUBMIT BUTTON TO SEND REQUEST TO THE HTTP PROGRAM TO GET THE COMPLETED HTML FORM
G
THE HTTP PROGRAM OF THE IPD SERVER PASSES THE ENTERED DATA SET (SELECTED FROM THE INPUT BOX) TO THE CGI AND THE CGI SCRIPT TRANSLATES THE QUERY STRING INTO A PROPER QUERY FOR SEARCHING THE RDBMS OF THE IPD SERVER IN THE SELECTED MODE
H
THE TRANSLATED QUERY IS USED TO SEARCH THE RDBMS TO FIND THE SET OF URLs RELATED TO THE REGISTERED CONSUMER PRODUCT ASSIGNED THE UPN IN THE SELECTED DATA TRIPLET
I
THE CGI SCRIPT CREATES HTML FORM, INSERTS RESULTS FROM SEARCH THEREINTO, AND SENDS THE HTML FORM TO THE REQUESTING CLIENT SUBSYSTEM
J
B

FIG. 6E2


B
DISPLAY THE SET OF URLs (FROM SEARCH) IN THE HTML FORM ON THE INFORMATION DISPLAY FRAME OF THE GUI OF REQUSTING CLIENT SUBSYSTEM
K
ACCESS AND DISPLAY ANY HTML DOCUMENT REFERENCED WITHIN THE DISPLAYED URL MENU BY SELECTING THE CORRESPONDING URL ANCHORED THERETO
L

F I G. 6E3

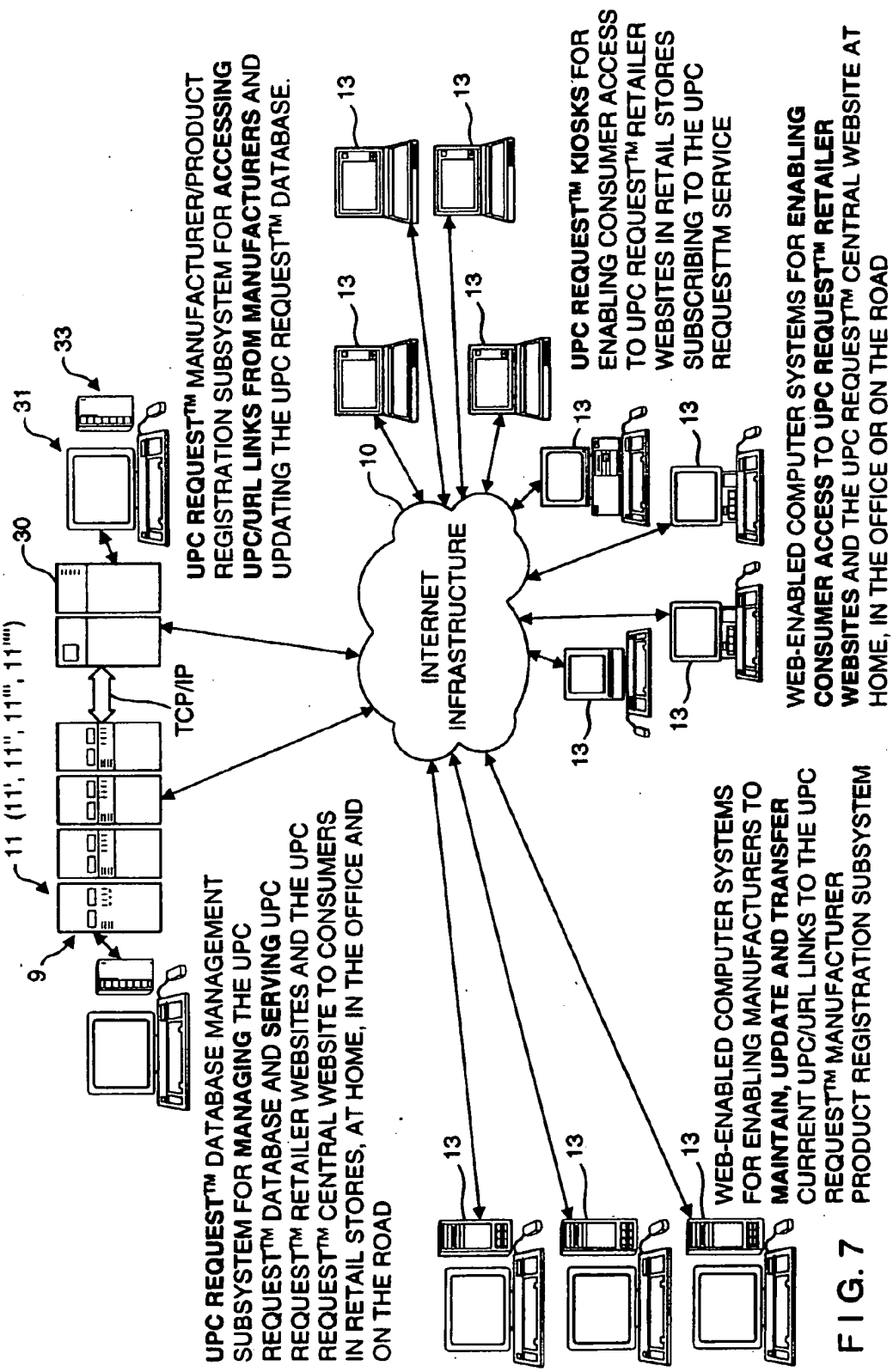

UPC REQUEST™ DATABASE MANAGEMENT SUBSYSTEM FOR MANAGING THE UPC REQUEST™ DATABASE AND SERVING UPC REQUEST™ RETAILER WEBSITES AND THE UPC REQUEST™ CENTRAL WEBSITE TO CONSUMERS IN RETAIL STORES, AT HOME, IN THE OFFICE AND ON THE ROAD
TCP/IP
UPC REQUEST™ MANUFACTURER/PRODUCT REGISTRATION SUBSYSTEM FOR ACCESSING UPC/URL LINKS FROM MANUFACTURERS AND UPDATING THE UPC REQUEST™ DATABASE.
INTERNET INFRASTRUCTURE
UPC REQUEST™ KIOSKS FOR ENABLING CONSUMER ACCESS TO UPC REQUEST™ RETAILER WEBSITES IN RETAIL STORES SUBSCRIBING TO THE UPC REQUESTTM SERVICE
WEB-ENABLED COMPUTER SYSTEMS FOR ENABLING MANUFACTURERS TO MAINTAIN, UPDATE AND TRANSFER CURRENT UPC/URL LINKS TO THE UPC REQUEST™ MANUFACTURER PRODUCT REGISTRATION SUBSYSTEM
WEB-ENABLED COMPUTER SYSTEMS FOR ENABLING CONSUMER ACCESS TO UPC REQUEST™ RETAILER WEBSITES AND THE UPC REQUEST™ CENTRAL WEBSITE AT HOME, IN THE OFFICE OR ON THE ROAD
9
11 (11', 11'', 11''', 11'''')
30
31
33
10
13

FIG. 7

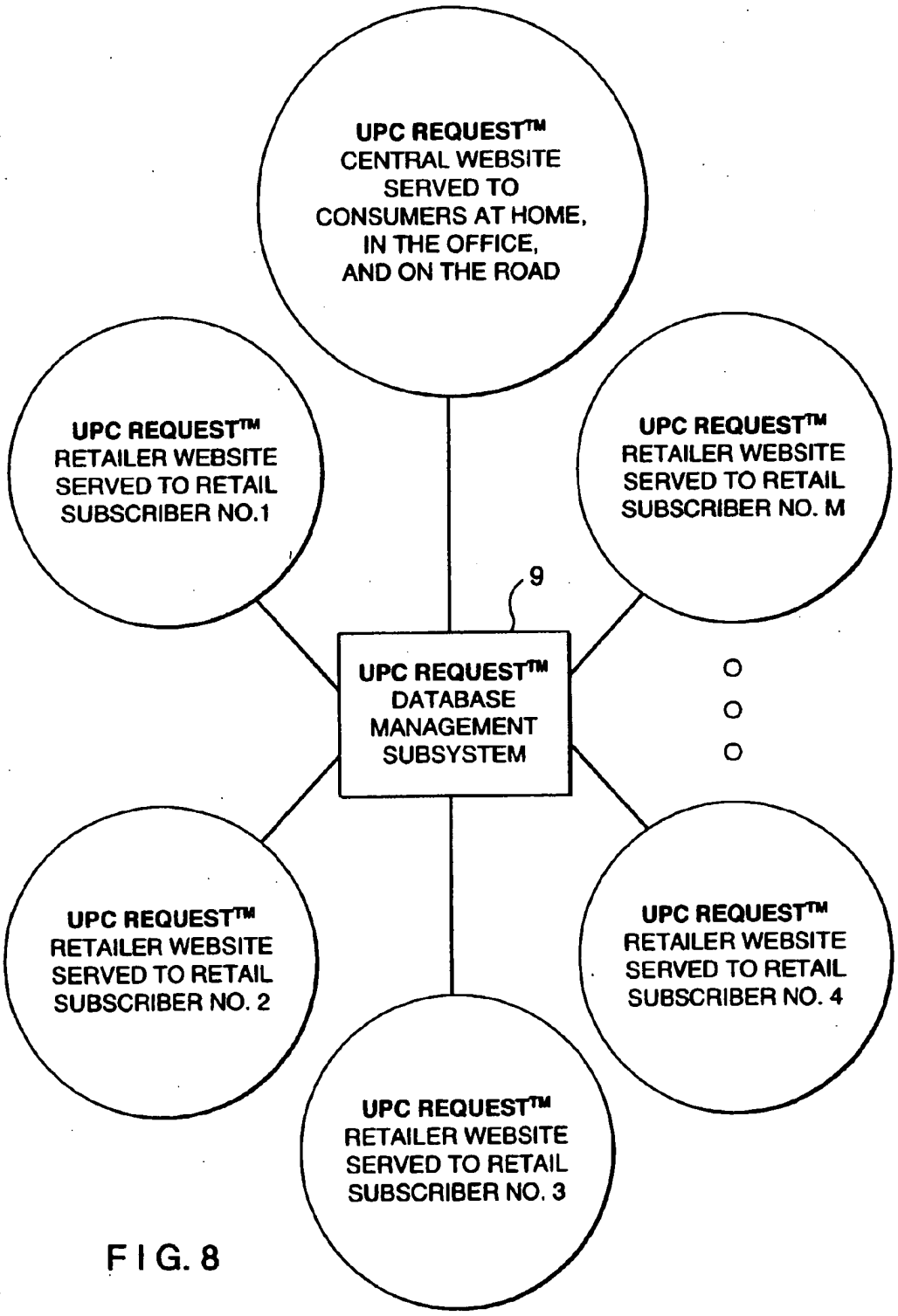

UPC REQUEST™ CENTRAL WEBSITE SERVED TO CONSUMERS AT HOME, IN THE OFFICE, AND ON THE ROAD
UPC REQUEST™ RETAILER WEBSITE SERVED TO RETAIL SUBSCRIBER NO.1
UPC REQUEST™ RETAILER WEBSITE SERVED TO RETAIL SUBSCRIBER NO. M
9
UPC REQUEST™ DATABASE MANAGEMENT SUBSYSTEM
UPC REQUEST™ RETAILER WEBSITE SERVED TO RETAIL SUBSCRIBER NO. 2
UPC REQUEST™ RETAILER WEBSITE SERVED TO RETAIL SUBSCRIBER NO. 4
UPC REQUEST™ RETAILER WEBSITE SERVED TO RETAIL SUBSCRIBER NO. 3

FIG. 8

METHOD OF AND SYSTEM FOR MANAGING AND SERVING CONSUMER-PRODUCT RELATED INFORMATION ON THE WORLD WIDE WEB (WWW) USING UNIVERSAL PRODUCT NUMBERS (UPNS) AND ELECTRONIC DATA INTERCHANGE (EDI) PROCESSES

RELATED CASES

[0001] This Application is a Continuation-in-Part of copending application Ser. No. 09/447,121 filed Nov. 22, 1999; copending application Ser. No. 09/441,973 filed Nov. 17, 1999; which is a Continuation-in-Part of copending application Ser. No. 09/284,197 which was entered into the U.S. on Apr. 21, 1999 which is a National Stage Entry Application from International Application No. PCT/US97/19227 filed Oct. 27, 1997, published as WIPO Publication No. WO 98/19259 on May 7, 1998; as well as a Continuation-in-Part of the following U.S. applications: Ser. No. 08/736,798 filed Oct. 25, 1996, Ser. No. 08/752,136 filed Nov. 19, 1996; Ser. No. 08/826,120 filed Mar. 27, 1997; Ser. No. 08/854,877 filed May 12, 1997, now U.S. Pat. No. 5,950,173; Ser. No. 08/871,815 filed Jun. 9, 1997; and Ser. No. 08/936,375 filed Sep. 24, 1997, each said Application is commonly owned by IPF, Inc., and is incorporated herein by reference in its entirety as if fully set forth herein.

BACKGROUND OF INVENTION

[0002] 1. Field Of Invention

[0003] The present invention generally relates to a novel system and method for collecting consumer-product related information and transmitting and delivering the same along the consumer-product supply and demand chain using the International Information Infrastructure (e.g. the Internet), and more particularly to a novel system and method for delivering consumer product related information to consumers within retail environments using Internet-based information servers and sales agents.

[0004] 2. Brief Description of the Prior Art

[0005] Dissemination of consumer-product information between manufacturers and their retail trading partners must be accurate and timely. The traditional methods of phone calls and faxes are time consuming and resource intensive. An electronic Universal Product Code (UPC) Catalog (i.e. database system), accessible 24 hours a day, is a solution. In 1988, Quick Response Services (QRS), Inc. Of Richmond, Calif., introduced the first independent product information database, using the retail industry standard UPC numbering system. Today the QRSolutions™ Catalog contains information on over 52 million products from over 1500 manufacturers. The QRSolutions Catalog is a Windows-based application providing a critical information flow link between the retailers and the manufacturers along the supply and demand chain.

[0006] After assigning a UPC number to each item, the manufacturer organizes and sends the data via an electronic data interchange (i.e. EDI) transmission, or a tape, to QRS, Inc. to be loaded into the UPC Catalog database. Changes to the data can be made on a daily basis. Retailers with access to a manufacturer's data can view and download the data once it has been added or updated. Automatic update capabilities ensure the most recent UPC data will be in the EDI mailbox of each retailer customer quickly.

[0007] The effect of a centralized database such as QRS's UPC Catalog improves the flow of merchandise from the manufacturer to the retailer's selling floor and ultimately to the consumer. With the UPC Catalog, accurate, up-to-date product information is available when the retailer needs it, eliminating weeks from the order cycle time.

[0008] In addition to the electronic UPC-based product information subsystem (i.e. UPC Catalog) described above, a number of other information subsystems have been developed for the purpose of providing solutions to problems relating to electronic commerce (EC) merchandising and logistics within the global supply chain. Such ancillary information subsystems include, for example: Sales, Analysis and Forecasting Subsystems for producing and providing retailers with information about what products consumers are buying; Collaborative Replenishment Subsystems for determining what products retailers can buy in order to satisfy consumer demand at any given point of time; and Transportation and Logistics Information Subsystems for producing and providing retailers with information about when products purchased by them (at wholesale) will be delivered to their stores. Typically, such information subsystems are connected to various value added information networks in order to efficiently offer such information services to retailers on a global basis.

[0009] While the above-described information systems collectively cooperate to optimize the process of moving raw materials into finished products and into the hands of consumers, such information systems fail to address the information needs of the consumers of retail products who require and desire product-related information prior to, as well as after, the purchase of consumer-products. Moreover, prior art demand chain management systems operate in an open-loop mode with a "break" in information flow cycle, disabling the manufacturers from communicating with the consumers in a n efficient manner to satisfy consumer needs.

[0010] Presently, an enormous amount of time, money and effort is being expended by companies in order to advertise and sell their products and services, and to provide product related information, product warranty service and the like after product purchase has taken place. Various types of media for decades have been used to realize such fundamental business functions.

[0011] In recent times, there has been a number of significant developments in connection with the global information network called the "Internet", which has greatly influenced many companies to create multi-media Internet Web-sites in order to advertise, sell and maintain their products and services. Examples of such developments include, for example: the World Wide Web (WWW) based on the Hypertext Markup Language (HTML) and the Hypertext Transmission Protocol (HTTP) by Tim Berners-Lee, et al. (See "World-Wide Web: The Information Universe" by Tim Berners-Lee, et al; easy to use Java GUI-based Internet navigation tools, such as the Netscape® D browser from Netscape Communications, Inc., the Internet Explorer™ browser from MicroSoft Corporation and the Mosaic™ browser from Spyglass Corporation; and the Virtual Reality Modeling Language (VRML) by Mark Pecse. Such developments in recent times have made it very easy for businesses to create 2-D Hypermedia-based Home Pages and 3-D VR Worlds (i.e. 3-D Web-sites) for the purpose of

projecting a desired "corporate image" and providing a backdrop for financial investment solicitation, as well as product advertising, sales and maintenance operations.

[0012] Presently, a person desiring to acquire information about any particular product has a number of available search options. In particular, he or she may attempt to directly contact the manufacturer, wholesaler or reseller by telephone, US mail, e-mail, or through the company's World Wide Web-site (WWW), if they have one. In the event one decides to acquire product information through the seller's WWW site, he or she must first determine the location of its WWW site (i.e. Internet address) which oftentimes can involve using Internet Search engines such as Yahoo®, AltaVista™, WebCrawler™, Lycos™, Excite™, or the like. This can be a very time consuming process and can sometimes lead to a dead end. Once the Internet address is obtained, one must then review the home page of the company's Web-site in order to find where information about a particular product resides on the Website, if it so exists. This search process can be both time consuming and expensive (in terms of Internet time) and may not turn up desired information on the product of interest.

[0013] In some instances, product brochures bear a preprinted Internet address designed to direct or point prospective customers to a particular Web-site where more detailed product information can be found. A recent example of this "preprinted Web Address" pointing technique is the 1996 product brochure published by the Sony Corporation for its Sony® PCV-70 Personal Computer, which refers prospective customers to the Sony Web Address "http://www.sony-.com/pc". While this approach provides a direct way of finding product related information on the Internet, it is not without its shortcomings and drawbacks.

[0014] In particular, when a company improves, changes or modifies an existing Web-site which publishes product and/or service advertisements and related information, it is difficult (if not impossible) not to change the Internet locations (e.g. Web addresses) at which such product and/or service advertisements and related information appear. Whenever a company decides or is forced to change any of its advertising, marketing and/or public relations firms, there is a substantial likelihood that new Web-sites will be created and launched for particular products and services, and that the Web addresses of such new Web-sites will no longer correspond with the Web addresses on preprinted product brochures currently in circulation at the time. This can result in pointing a consumer to erroneous or vacant Web-sites, that present either old or otherwise outdated product and/or service information, possibly adversely influencing the consumer's purchasing decision.

[0015] Moreover, when a company launches a new Web-site as part of a new advertising and marketing campaign for a particular product, any preprinted advertising or marketing material relating to such products will not reflect the new Web-site addresses which the campaign is promoting. This fact about preprinted advertising media renders it difficult to unify new and old advertising media currently in circulation into an advertising and marketing campaign having a coherent theme. In short, the inherently static nature of the "preprinted Web address" pointing technique described above is wholly incapable of adjusting to the dynamic needs of advertising, marketing and public relations firms alike.

[0016] Recently, two different methods have been proposed for providing product information to consumers over the Internet.

[0017] U.S. Pat. No. 5,640,193 to Wellner discloses a system and method for accessing and displaying Web-based consumer product related information to consumers using a Internet-enabled computer system, whereby in response to reading a URL-encoded bar code symbol on or associated with a product, the information resource specified by the URL is automatically accessed and displayed on the Internet-enabled computer system. While this system and method enables access of consumer product information related information resources on the WWW by reading URL-encoded bar code symbols, it requires that custom URL-encoded bar code symbols be created and applied to each and every consumer product in the stream of commerce.

[0018] U.S Pat. No. 5,978,773 to Hudetz, et al discloses a solution to the problem presented by the system and method of U.S Pat. No. 5,640,193, by proposing the use of a UPC/URL database in order to translate UPC numbers read from consumer products by a bar code scanner, into the URLs of published information resources on the WWW relating to the UPC-labeled consumer product.

[0019] Like U.S Pat. No. 5,978,773 to Hudetz, et al, WIPO Publication No. WO 98/03923 discloses the use of a UPC/URL database in order to translate UPC numbers read from consumer products by a bar code scanner, into the URLs of published information resources on the WWW relating to the UPC-labeled consumer product.

[0020] While U.S Pat. No. 5,978,773 and WIPO Publication No. WO 98/03923 both provide an effective solution to the problem presented by U.S Pat. No. 5,640,193, these prior art references completely fail to recognize or otherwise address the myriad of problems relating to UPC/URL-link collection, management, delivery, access and display along the retail supply and demand chain, which must be first solved in order deliver a technically feasible, globally-extensive, UPC-driven consumer product information system for the benefit of consumers worldwide.

[0021] Thus, it is clear that there is great need in the art for an improved Internet-based method of and system for delivering product related information to the consumers along the entire retail supply and demand chain, while avoiding the shortcomings and drawbacks of prior art systems and methodologies.

DISCLOSURE OF THE INVENTION

[0022] Accordingly, a primary object of the present invention is to provide a novel method and apparatus for collecting product-related information and transmitting and delivering the same between the manufacturers and retailers of products to the consumers thereof in retail shopping environments as well as at home, work and on the road, while overcoming the shortcomings and drawbacks of prior art systems and methodologies.

[0023] Another object of the present invention is to provide such apparatus in the form of a novel consumer-product information collection, transmission and delivery system.

[0024] Another object of the present invention is to provide such a system with an Internet-based product informa-

tion database subsystem which, for each commercially available consumer-product, stores a number of information elements including: the name of the manufacturer; the Universal Product Code (UPC) assigned to the product by the manufacturer; one or more URLs specifying the location of information resources (e.g. Web-pages) on the Internet relating to the UPC-labeled consumer-product; and the like.

[0025] Another object of the present invention is to provide such a system, in which the URLs stored in the Internet-based product information database are categorically arranged and displayed according to specific types of product information (e.g., product specifications and operation manuals; product wholesalers and retailers; product advertisements and promotions; product endorsements; product updates and reviews; product warranty/servicing; related or complementary products; product incentives including rebates, discounts and/or coupons; etc.) that relate to the kind of information required, desired or otherwise sought by consumers, wholesalers, retailers and/or trading partners; product prices at which the products are being offered for sale by a particular retailer; and the like.

[0026] Another object of the present invention is to provide such a system, wherein the information maintained within the Internet-based product information database management subsystem provides a manufacturer-defined consumer-product directory that can be used by various persons along the retail supply and demand chain.

[0027] Another object of the present invention is to provide such a system, wherein the manufacturers of consumer-products are linked to the retailers thereof in the middle of the supply and demand chain by allowing either trading partner to access to consumer-product information from the Internet-based product information database virtually 24 hours a day, seven days a week.

[0028] Another object of the present invention is to provide such a system, wherein consumer-product manufacturers, their advertisers, distributors and retailers are linked to the consumers of such products at the end of the supply and demand chain, by allowing such parties access to consumer-product information from the Internet-based product information database subsystem virtually 24 hours a day, seven days a week.

[0029] Another object of the present invention is to provide such a system an d method of using the same, which will accelerate the acceptance of electronic commerce on the Internet and the development of the electronic marketplace, which can be used by consumers and small and large businesses alike.

[0030] Another object of the present invention is to provide a novel system and method for finding and serving consumer-product related information on the Internet.

[0031] Another object of the present invention is to provide such a system and method, wherein virtually any type of product can be registered with the system by symbolically linking or relating (i) its preassigned Universal Product Number (e.g. UPC or EAN number) or at least the Manufacture Identification Number (MIN) portion thereof with (ii) the Uniform Resource Locators (URLs) of one or more information resources on the Internet (e.g. the home page of the, manufacturer's Web-site) related to such products.

[0032] Another object of the present invention is to provide such a system and method wherein a Web-based document transport subsystem is provided for use by manufacturers as well as their advertisers and agents in registering the UPNs (e.g. UPC numbers) of their products and the URLs of the information resources related to such products.

[0033] Another object of the present invention is to provide such a system with a number of different modes of operation, namely: a Manufacturer/Product Registration Mode, wherein manufacturers can register their companies and consumer products (e.g. UPC numbers and URLs) with the system; an UPN-Directed Information Access Mode, wherein consumers can access and display information menus containing UPC numbers linked to URLs pointing Web pages containing consumer product related information by scanning the UPC label on the consumer product or by entering the UPC number thereof into a data-entry screen displayed by the system in this mode; a Manufacturer Website Search Mode, wherein the home page of a manufacturer's Web-site can be automatically accessed and displayed by scanning the UPC label on any consumer product of the manufacturer or by entering the UPC number thereof into a data-entry screen displayed by the system in this mode; a Trademark-Directed Search Mode enabling consumers to use trademarks and/or tradenames associated with consumer products to search for consumer-product related information registered within the system; and a Product-Description Directed Search Mode enabling consumers to use product descriptors associated with particular consumer products to search for consumer-product related information registered within the system.

[0034] Another object of the present invention is to provide such a system, wherein when the system is in its UPN-Directed Information Menu Access Mode, a predesignated information resource (e.g. advertisement, product information, etc.) pertaining to any commercial product registered with the system can be automatically accessed from the Internet and displayed from the Internet browser by simply entering the registered product's UPN into the Internet browser manually or by bar code symbol scanning.

[0035] Another object of the present invention is to provide such a system, wherein when the system is in its Trademark-Directed Search Mode, a predesignated information resource (e.g. advertisement, product information, etc.) pertaining to any commercial product registered with the system can be automatically accessed from the Internet and displayed from the Internet browser by simply entering the registered product's trademark(s) and/or associated company name into the Internet browser.

[0036] Another object of the present invention is to provide such a system, wherein when the system is in its Product-Description-Directed Search Mode, a predesignated information resource (e.g. advertisement, product information, etc.) pertaining to any commercial product registered with the system can be automatically accessed from the Internet and displayed from the Internet browser by simply entering the registered product's product description into the Internet browser.

[0037] Another object of the present invention is to provide such a system, wherein a predesignated information resource pertaining to any commercial product having been assigned a Universal Product Number (UPN) can be

accessed from the Internet and displayed from the Internet browser by simply selecting its and then entering the UPN numeric string into an Input Box which pops up on an HTML form displayed by an Internet browser.

[0038] Another object of the present invention is to provide such a system in which a relational database, referred to as "an Internet Product Directory (IPD)," is realized on one or more data-synchronized IPD Servers for the purpose of registering product related information, namely: (i) information representative of commercial product descriptions, the trademarks used in connection therewith, the company names providing and/or promoting such products, the E-mail addresses of such companies, and the corresponding URLs on the Internet specifying current (i.e. up-to-date) Internet Web-site locations providing product-related information customized to such products.

[0039] Another object of the present invention is to provide such a product information finding and serving system, wherein the URLs symbolically linked to each registered product in the IPD Servers thereof are categorized as relating primarily to Product Advertisements, Product Specifications, Product Updates, Product Distributors, Product Warranty/Servicing, and/or Product Incentives (e.g. rebates, discounts and/or coupons), and that such URL categories are graphically displayed to the requester by way of easy-to-read display screens during URL selection and Web-site connection.

[0040] A further object of the present invention is to provide an Internet-based System wherein: (1) manufacturers and their agents are enabled to simply link (i.e. relate), manage and update within a centralized database, the UPC (and/or UPC/EAN) numbers on their products and the Uniform Resource Locators (URLs) of HTTP-encoded document (i.e. Web pages) containing particular kinds of consumer product-related information published on the Internet by the manufacturers, their agents and/or third parties; and (2) consumers, in retail stores, at home, in the office and on the road, are enabled to simply access such consumer product-related information using such UPC (and/or UPC/EAN) numbers and/or by scanning UPC (or UPC/EAN) bar code symbols encoded with such product identification numbers.

[0041] Another object of the present invention is to provide a novel method of carrying out electronic-type commercial transactions involving the purchase of products which are advertised on the Internet at uniform resource locations (URLs) that are registered with the IPI system of the present invention.

[0042] Another object of the present invention is to provide a novel system and method of finding the UPN or USN associated with any particular registered product, respectively, by simply selecting a Java GUI button on the Internet browser display screen in order to enter a "Trademark-Directed Search Mode", whereby (i) a dialogue box is displayed on the display screen requesting any known trademarks associated with the product, and/or the name of the company that makes, sells or distributes the particular product, and (ii) the corresponding UPN (i.e. UPC number or EAN number) registered with the IPD Servers is displayed to the user for acceptance, whereupon the Internet Uniform Resource Locators (URLs) are automatically accessed from the IPD Servers and displayed on the display screen of the Internet browser for subsequent URL selection and Web-site connection.

[0043] Another object of the present invention is to provide such a system and method, wherein during the Trademark-Directed Search Mode, the UPN (e.g. UPC or EAN number) associated with any registered product can be found within the database of the IPD Server using any trademark(s) and/or the company name commonly associated with the product.

[0044] Another object of the present invention is to provide a novel system and method for serving consumer-product related information to Internet users in retail shopping environments (e.g. department stores, supermarkets, superstores, home-centers and the like) as well as at home, work or on the road.

[0045] Another object of the present invention is to provide a consumer product information access terminal located at a point-of-sale (POS) station, wherein the bar code symbol reader integrated with the POS station can be used to read the UPC numbers on consumer products being offered for sale in the store in order to access consumer product related information from hyper-linked Web-sites on the Internet, for display on an LCD screen located at the POS station and viewable from various positions by the sales clerk as well as consumer shoppers.

[0046] Another object of the present invention is to provide a system and method, wherein one or more computer-based kiosks are installed within retail shopping environments and each such kiosk has an automatic bar code symbol reader for reading the UPC numbers on consumer products being offered for sale in the store, and also an LCD touch-type display screen for displaying product-related information accessed from hyper-linked Web-sites on the Internet.

[0047] Another object of the present invention is to provide such a system and method, wherein one or more computer-based kiosks are installed within retail shopping environments and each such kiosk has an automatic bar code symbol reader for reading the UPC numbers on consumer products being offered for sale in the store, and also a LCD touch-type display screen for displaying product-related information accessed from hyper-linked Web-sites on the Internet.

[0048] Another object of the present invention is to provide such a consumer product information kiosk, wherein the laser scanning bar code symbol reader can be easily removed from its support stand to scan large consumer products that might be difficult to present within the scanning field while the bar code symbol reader is supported above the LCD display panel.

[0049] Another object of the present invention is to provide such a consumer product information kiosk, wherein the laser scanning bar code symbol reader has a cordless interface with the kiosk so that it may be moved about within a retail store in a portable manner to scan UPC labels and access consumer product related information.

[0050] Another object of the present invention is to provide a consumer product information kiosk for use with the system hereof, that is completely transportable within the

store by hand, or may be mounted upon a shopping cart or other vehicle for the convenience of shoppers and the like.

[0051] Another object of the present invention is to provide "virtual" or "Cyber" sales and service agents within retail shopping environments by installing the computer-based kiosks of the present invention therein.

[0052] Another object of the present invention is to provide a Web-based information delivery system and method, wherein the computer-based kiosks employed throughout the hosting retailer's store are capable of displaying the price of products offered for sale in the store upon reading the UPC bar code symbol thereon.

[0053] Another object of the present invention is to provide a novel method of constructing a relational database for use within the product information finding and serving subsystem of the present invention.

[0054] Another method of the present invention is to provide such a method of database construction, wherein the relational database is initially "seeded" with: (i) the Manufacturer Base UPC Numbers based on the six digit UPC Manufacturer Identification Numbers (MIN) assigned to the manufacturers by the UCC and incorporated into the first six characters of each UPC number applied to the products thereof; and (ii) the URLs of the Web-site home pages of such manufacturers.

[0055] Another method of the present invention is to provide such a method of database construction, wherein the "seeded" relational database is then subsequently extended and refined with the participation of each registered manufacturer (and/or agents thereof) by adding to the "seeded" database (iii) the 12 digit UPC numbers assigned to each product sold thereby and the menu of URLs symbolically linked to each such corresponding product.

[0056] Another object of the present invention is to provide such a system and method, in which Web-site-based advertising campaigns can be changed, modified and/or transformed in virtually any way imaginable by simply restructuring the symbolic links between the products and/or services in the campaign using current (i.e. up-to-date) Web-site addresses at which Web-site advertisements and information sources related thereto are located on t the Internet.

[0057] Another object of the present invention is to provide a novel system and method of automatically soliciting companies to register their products within the databases of such IPD Servers in order that product related information of a multimedia nature (e.g. Web-sites), once registered therewith, can be easily found on the Internet by anyone using the system and method of the present invention.

[0058] Another object of the present invention is to provide a novel system and method for finding and serving consumer-product related information on the Internet, accessible from the Websites of each manufacturer who has registered its UPN/URLs with the system's "central" IPD Database.

[0059] Another object of the present invention is to provide such a system and method, wherein as part of the consumer product registration process, the manufacturer (or retailer) maintains a limited-version of the UPN/URL database which contains a list of categorized URLs for each UPC-encoded product that it sells.

[0060] Another object of the present invention is to provide such a system and method, wherein the consumer product related information links contained within the limited-version of the UPN/URL Database of each registered manufacturer (or retailer) can be accessed from the manufacturer's (or retailer's) company Website and served to consumers requesting such information by way of UPC (or EAN) number entry.

[0061] Another object of the present invention is to provide such a system and method, wherein input HTML forms for searching the limited-version of the UPN/URL Database of each registered manufacturer (or retailer) can be accessed by selecting a predesignated Check Button on the Java GUI to a Website providing access thereto.

[0062] Another object of the present invention is to provide such a system and method, wherein the limited-version of the UPN/URL Database of each registered manufacturer (or retailer) is used to update a "central" or "master" UPN/URL Database which is continuously maintained and made accessible to consumers (i) through Web-based kiosks installed in licensed retail environments and (ii) through Internet-enabled client subsystems located at home, work and school.

[0063] Another object of the present invention is to provide such a system and method, wherein at the time of registering each manufacturer (or retailer) with the system, an Internet-based registration server automatically transmits a computer program to the manufacturer's (or retailer's) computer system for use in constructing and maintaining the limited-version UPN/URL Database related to the manufacturer's (retailer's) products.

[0064] Another object of the present invention is to provide such a system and method, wherein the limited-version UPN/URL Database of each registered manufacturer (or retailer) can be served from the manufacturer's (or retailer's) Internet information server using a Common Gateway Interface program (CGI) Another object of the present invention is to provide such a system and method, wherein the limited-version UPN/URL database of each registered manufacturer (or retailer) is realized using a cross-platform compatible, relational database management system (RDBMS).

[0065] Another object of the present invention is to provide a novel method of and system for accessing consumer product related information at points within HTML-encoded documents, at which Universal Product Number (UPN) encoded Java Applets are embedded so as to produce, when executed, a consumer product information display enabling ("CPID-enabling") Java-based graphical user interfaces (GUIs) for the convenience of consumers shopping at electronic-commerce (EC) enabled stores, considering the placement of bids at on-line auction sites, or browsing product advertisements appearing on the World Wide Web.

[0066] Another object of the present invention is to provide an Internet-based consumer product information collection, managing and delivery system and method, wherein for each consumer product registered within the UPN/URL database of the system, there is created and stored, an interactive consumer product information request (CPIR)

enabling Applet (e.g. based on Java™ component principles or MicroSoft's Active-X technology) which, when executed upon the initiation of the consumer through a mouse-clicking operation, automatically causes a preassigned CPID-enabling Java GUI to be displayed at the consumer's point of presence in Cyberspace, revealing the results of a consumer product information display conducted upon the product identified by the UPN encoded within the executed Applet.

[0067] Another object of the present invention is to provide such an Internet-based consumer product information collection, managing and delivery system and method, wherein (1) the UPN assigned to a particular consumer product by the manufacturer and (2) the URL of the Java script running on the IPD server of the system are encoded within the CPIR-enabling Applet so that, upon execution of the Applet, a consumer product information display (CPID) Java GUI is automatically produced for the consumer's convenience.

[0068] Another object of the present invention is to provide an Internet-based consumer product information collection, managing and delivery system and method, wherein the CPID-enabling Java GUI automatically displays a manufacturer-defined menu (i.e. list) of categorized URLs pointing to information resources on the Internet (e.g. WWW) relating to the consumer product identified by the UPN encoded within the CPIR-enabling Applet.

[0069] Another object of the present invention is to provide such an Internet-based consumer product information collection, managing and delivery system and method, wherein CPIR-enabling Applets are created by the system administrator, loaded within the UPN/URL database management subsystem thereof, distributed to retailers, wholesalers, manufacturers, advertisers and others for embedding within HTML-encoded documents associated with EC-enabled stores, catalogs, Internet-based product advertisements, on-line auction sites, and other locations on the WWW where accurate consumer product related information is desired or required without leaving the point of presence on the WWW at which the consumer resides.

[0070] Another object of the present invention is to provide such an Internet-based consumer product information collection, managing and delivery system and method, wherein CPIR-enabling Applets are created, distributed, embedded within a HTML-encoded document related to a particular consumer product, and subsequently executed by a consumer so as to access and display a manufacturer-defined menu (i.e. list) of categorized URLs pointing to product-related Web-documents.

[0071] Another object of the present invention is to provide such an Internet-based consumer product information collection, managing and delivery system and method, wherein the consumer initiating the execution of a particular CPIR-enabling Applet may be anyone desiring or requiring consumer product related information while interacting with the communication medium provided by the Internet and its supported technologies (e.g. WWW, EC, etc.). As such, the consumer may be a student shopping at an EC-enabled (business-to-consumer) retail store for textbooks, a retail purchasing agent shopping at an on-line (business-to-business) wholesale product catalog for product inventory, a dealer looking to purchase a new or used product listed at an online auction site, or anyone encountering an Internet-based advertisement while surfing the WWW.

[0072] Another object of the present invention is to provide such an Internet-based consumer product information collection, managing and delivery system and method, wherein a thumb-nail picture, arbitrary graphical object, predesignated CPIR-indicating icon, or hypertext-type link associated with a particular consumer product can be embedded within the CPIR-enabling Applet associated therewith, so as to enable the consumer to produce a CPID-enabling Java GUI upon encountering the same in an HTML-encoded document on the WWW.

[0073] Another object of the present invention is to provide such an Internet-based consumer product information collection, managing and delivery system and method, wherein licensed users can download CPIR-enabling Applets from the CPIR-Enabling Applet Library to any client computer for eventual insertion within the HTML code of a particular Web-document to be published on the Internet in accordance with the licensing arrangement between the contracting parties. Such end-use applications might be in EC-enabled retail product catalogs, EC-enabled wholesale/trade catalogs, Internet-based product advertisements, on-line auction WWW sites, on-line stock trading WWW sites, and the like.

[0074] Another object of the present invention is to provide such an Internet-based consumer product information collection, managing and delivery system and method, wherein the CPID-enabling Java GUIs enabled by executed CPIR-enabling Applets can function as CPI-serving "cyber-kiosks" that can be installed at any location in Cyberspace for the convenience of consumers residing therewithin without disturbing their point of presence.

[0075] Another object of the present invention is to provide a novel method of and system for delivering consumer product advertisements and consumer product related information to consumers over the WWW involving the use of a single mouse-clicking operation by the consumer.

[0076] Another object of the present invention is to provide a novel method of and system for embedding CPIR-enabling Applets within HTML-encoded consumer product advertisements published over the WWW involving the use of a single mouse-clicking operation by the consumer.

[0077] Another object of the present invention is to provide a novel method of and system for delivering consumer product related information to consumers at on-line auction sites on the WWW involving the use of a single mouse-clicking operation by the consumer.

[0078] Another object of the present invention is to provide a novel method of and system for embedding CPIR-enabling Applets within HTML-encoded on-line auction pages published over the WWW.

[0079] Another object of the present invention is to provide a novel method of and system for embedding CPIR-enabling Applets within HTML-encoded securities performance charts published at on-line electronic securities trading site on the WWW.

[0080] Another object of the present invention is to provide a new method of and system for purchasing a consumer product over the Internet (e.g. WWW) comprising the steps

of: embedding a UPN-encoded CPIR-enabling Applet within the HTML-code of a consumer product advertisement, wherein the CPIR-enabling Applet when executed displays a categorized URL menu containing one or more URLs pointing to one or more EC-enabled stores or on-line catalogs on the WWW at which the consumer product identified by the encoded UPN can be purchased and delivered to a particular address in physical space.

[0081] Another object of the present invention is to provide a novel cyber-kiosk, launchable from predefined points of presence within an EC-enabled store, on-line product catalog or other type of WWW site, for enabling consumers (including retail purchasing agents) to quickly access and display at the predefined point of presence, an interactive menu of categorized URLs pointing to consumer product related information resources published on the WWW and symbolically linked to the UPNs of consumer products within a centralized UPN/URL database management subsystem, by manufacturers and/or their agents.

[0082] Another object of the present invention is to provide a novel EC-enabled product catalog having a library of CPIR-enabling Applets embeddable within graphical images of consumer products in HTML-encoded documents and enabling, when executed, a UPN-directed search within the UPN/URL database management subsystem and the display of an interactive menu of categorized URLs pointing to consumer product related information resources published on the WWW and symbolically linked to the UPNs of consumer products within a centralized UPN/URL database management subsystem, by manufacturers and/or their agents.

[0083] Another object of the present invention is to provide a novel Internet-based electronic commerce (EC) enabled shopping system comprising an Internet information server connected to the infrastructure of the Internet and supporting the hypertext transmission protocol (http), a Web-enabled client subsystem connected to the infrastructure of the Internet, an EC-enabled WWW site comprising a plurality of interlinked HTML-encoded documents arranged and rendered to provide an electronic store environment when served to a consumer operating the Web-enabled client subsystem, wherein the electronic store environment presents a plurality of products for purchase and sale by an EC-enabled payment method supported over the Internet.

[0084] Another object of the present invention is to provide such an Internet-based electronic commerce (EC) enabled shopping system, wherein a Java Applet tag, associated with each product, is embedded within at least one of the HTML-encoded documents displayed on the Web-enabled client subsystem, and each Java Applet tag is associated with a Java Applet encoded with the universal product number (UPN) assigned to one of the products, and, when the consumer selects one of the Java Applet tags, the associated Java Applet is automatically executed enabling a search to be conducted against a product information database hosted on an Internet database server connected to the Internet, from which the results of the UPN-specified search are automatically displayed in a GUI served to the Web-enabled client subsystem.

[0085] Another object of the present invention is to provide client-side and server CPIR-enabling Java Applets for enabling the consumer product information searches at virtually any consumer point of presence on the WWW by performing a single mouse-clicking operation.

[0086] Another object of the present invention is to provide a novel Internet-based system and method, wherein a plurality of publisher-operated client subsystems (i.e. manufacturer-operated client subsystems) are connected to a local or wide area TCP/IP-based network, for the purpose of enabling different departments within the publishing organization (e.g. advertising, world news, business, technology, sports, finance, education, arts and leisure, etc.) manage different types of UPN/URL links based on the type of information contained within the URL-specified information resource on the WWW.

[0087] Another object of the present invention is to provide a novel Internet-based system and method, wherein each publisher is provided with a computer-based publishing system, which enables the electronic layout of: (i) a Web-based publication (e.g. expressible in HTML or SGML code) having different content and advertising sections associated with each Web-page thereof and each such Web-page being located on the WWW at a particular URL; and (ii) a print-media based publication (e.g. expressible in a desired font) having different content sections and advertising sections associated with each printed-page thereof, wherein each such content section and advertising section is assigned a Universal Product Number (UPN) which is symbolically linked to a particular content or advertising section on the corresponding Web-page.

[0088] Another object of the present invention is to provide a novel Internet-based system and method, wherein the computer-based publishing subsystem is capable of automatically generating UPN/URL data link tables listing the URLs of each Web page symbolically linked to UPN assigned to a corresponding printed media page, and that such UPN/URL data link tables are transportable to a UPN/URL database management subsystem using electronic data interchange techniques, thereby enabling consumers (e.g. readers) to link from print-media to corresponding Web-based media using the UPNs printed on documents and the like only moments after the Web and print publications have been approved for publishing and sent to a http server and printing press, respectively.

[0089] Another object of the present invention is to provide a novel Internet-based consumer product information system and method for use in retail shopping environments, wherein each Web-enabled bar code driven consumer product information kiosk deployed therewithin embodies e-mail messaging capabilities which enable consumers to automatically save and link CPI-related Web documents as individual attachments to a preformatted e-mail message that is transmitted from a retailer-operated e-mail server, to a remote e-mail address specified by the consumer within the retail shopping environment.

[0090] Another object of the present invention is to provide a novel Internet-based consumer product information system and method for use in retail shopping environments, wherein each Web-enabled bar code driven consumer product information kiosk deployed therewithin embodies e-mail messaging capabilities which enable consumers to automatically save and record the URLs of CPI-related Web documents within the message field of a preformatted e-mail message that is transmitted from a retailer-operated e-mail

server, to a remote e-mail address specified by the consumer within the retail shopping environment.

[0091] Another object of the present invention is to provide such novel Internet-based consumer product information system and method for use in retail shopping environments, wherein the e-mail envelope is addressed with the consumer/shopper's home, office or like e-mail address by either reading an e-mail address encoded within a bar code (or magnetic-stripe) structure or manually entering the same within the addressee field, and the stuffed e-mail envelope is transported to its destination by manual selection of a "send" button within the displayed e-mail envelope.

[0092] Another object of the present invention is to provide such a novel Internet-based consumer product information system and method, wherein one or more central e-mail servers are used to collect copies of e-mail documents (and records thereof) transmitted from the Web/e-mail enabled kiosks within each retail shopping environments, for consumer and demographic information analysis, compilation, and storage within RDBMSs that are made accessible to retailers and manufacturers alike for use in product marketing, sales forecasting, customer intelligence, and like operations which enable more effective marketing of consumer products and services in both physical and electronic forms of commerce.

[0093] These and other objects of the present invention will become apparent hereinafter and in the Claims to Invention

BRIEF DESCRIPTION OF THE DRAWINGS

[0094] For a more complete understanding of how to practice the Objects of the Present Invention, the following Detailed Description of the Illustrative Embodiments can be read in conjunction with the accompanying Drawings, wherein:

[0095] **FIG. 1** is a schematic diagram illustrating the various information subsystems provided by the consumer-product information collection, transmission and delivery system of invention along the consumer-product demand chain, namely an Internet-based Product-Information (IPI) Finding and Serving Subsystem, a UPC-based Product-Information Subsystem ("UPC Catalog"), an Electronic Trading Information Subsystem, a Sales Analysis and Forecasting Information Subsystem, Collaborative Replenishment Information Subsystem, and a Transportation and Logistics Information Subsystem;

[0096] FIGS. **2-1** and **2-2** show a schematic diagram of the consumer-product information collection, transmission and delivery system of the illustrative embodiment hereof shown embedded with the infrastructure of the global computer communications network known as the "Internet", and comprising a plurality of data-synchronized Internet Product Directory (IPD) Servers connected to the infrastructure of the Internet, a UPN/URL Database Management Subsystem (i.e. UPN/URL Database) connected to one or more of the IPD Servers and one or more globally-extensive electronic data interchange (EDI) networks, a Web-based Document Server connected to at least one of the IPD Servers and the Internet infrastructure, a Web-based Document Administration Computer connected to the Web-based Document Server by way of a TCP/IP connection, a plurality of manufacturer-related electronic-commerce (EC) information servers for hosting EC-enabled stores or EC-enabled on-line catalogues of manufacturers, a plurality of retailer-related electronic-commerce (EC) information servers for hosting EC-enabled stores or EC-enabled on-line catalogues of retailers, a plurality of Internet Product-Information (IPI) Servers connected to the infrastructure of the Internet for serving consumer-product related information to consumers in retail stores and at home, a central e-mail RDBMS for receiving and storing copies of e-mail transmissions from retailer-store based kiosks to e-mail addresses of consumer accessing consumer product information therewith in retail shopping environments, a plurality of Client Subsystems connected to the infrastructure of the Internet and allowing manufacturers to transmit consumer-product related information to the Web-based Document Server for collection and retransmission to the IPD Servers, and a plurality of Client Subsystems connected to the infrastructure of the Internet and allowing consumers in retail stores and at home to request and receive consumer-product related information from the IPD Servers;

[0097] **FIG. 2A** is a schematic diagram illustrating the flow of information along the consumer-product supply and demand chain, including (i) the communication link extending between the information subsystems of manufacturers of UPC-encoded products and the centralized (or master) UPN/URL Database Management Subsystem **9** of the consumer-product information collection, transmission and delivery system of the present invention, (ii) the communication link extending between the UPN/URL Database Management Subsystem and the IPD Servers of the present invention, (iii) the communication link extending between the IPD Servers and in-store Client Subsystems of retailers, (iv) the communication link extending between the IPI Servers and the in-store Client Subsystems of retailers, (v) the communication link extending between the IPD Servers and the Client Subsystems of consumers, (vi) the communication link extending between the IPI Servers and the Client Subsystems of consumers, and (vii) the communication link extending between the UPN/URL Database Management Subsystem and the EC-enabled UPN-based Consumer Product Catalogue Server(s) of the present invention for providing consumer product catalogue services to retailer purchasing agents and others and enabling the on-line purchase of consumer products between trading partners (e.g. manufactures and retailers) using EDI (or XML/EDI) based business-to-business electronic commerce transactions;

[0098] **FIG. 2B1** is a block schematic diagram of the IPD Server of the first illustrative embodiment, showing its subsystem components namely a relational database management subsystem (RDBMS) server and a Java Web Server with Java servlet support, being accessed by a Java-enabled client machine seeking to access consumer product related information from the RDBMS server using server-side Java Applets whose HTML tags are embedded within HTML-encoded documents served to the client machine from any one of a number of potential http information servers on the Internet;

[0099] **FIG. 2B2** is a block schematic diagram of the IPD Server of the second illustrative embodiment, showing its subsystem components namely a relational database management subsystem (RDBMS) server and a Java Web Server with CGI script support, being accessed by a Java-enabled

client machine seeking to access consumer product related information from the RDBMS server using client-side Java Applets whose HTML tags are embedded within HTML-encoded documents served to the client machine from any one of a number of potential http information servers on the Internet;

[0100] FIG. 2B3 is a block schematic diagram of the IPD Server of the third illustrative embodiment, showing its subsystem components, namely: a relational database management subsystem (RDBMS) server and a Java Web Server being accessed by a Java-enabled client machine seeking to access consumer product related information from the RDBMS server using (i) a socket connection between the client machine and the Java Web server and (ii) client-side Java Applets whose HTML tags are embedded within HTML-encoded documents served to the client machine from any one of a number of potential http information servers on the Internet;

[0101] FIG. 2B4 is a block schematic diagram of the IPD Server of the fourth illustrative embodiment, showing its subsystem components, namely: a relational database management subsystem (RDBMS) server and a Java Web Server being accessed by a Java-enabled client machine seeking to access consumer product related information from the RDBMS server using (i) a Remote Method of Invocation (RMI) and (ii) client-side Java Applets whose HTML tags are embedded within HTML-encoded documents served to the client machine from any one of a number of potential http information servers on the Internet;

[0102] FIG. 2C is a schematic representation of a portion of the system shown in FIGS. 2-1 and 2-2, wherein a plurality of manufacturer-operated client subsystems are shown connected to a local or wide area IP-based network, preferably maintained behind a secure corporate firewall, and the secured manufacturer information network is connected to the infrastructure of the Internet by way of an Internet router and server, for the purpose of enabling different departments within a business organization (e.g. marketing, sales, engineering, support and service, advertising, finance, etc.) manage different types of UPN/URL links based on the type of information contained within the URL-specified information resource on the WWW;

[0103] FIG. 2D is a schematic representation of a portion of the system shown in FIGS. 2-1 and 2-2, wherein a plurality of publisher-operated client subsystems (i.e. manufacturer-operated client subsystems) are shown connected to a local or wide area IP-based network, preferably maintained behind a secure corporate firewall, and the secured manufacturer information network is connected to the infrastructure of the Internet by way of an Internet router and server, for the purpose of enabling different departments within the publishing organization (e.g. advertising, world news, business, technology, sports, finance, education, arts and leisure, etc.) manage different types of UPN/URL links based on the type of information contained within the URL-specified information resource on the WWW;

[0104] FIG. 2E1 is a schematic representation of the split-screen GUI associated with the computer-based publishing software program of the present invention operated in its composition/editorial mode, and having integrated UPN/URL data linking functionalities, showing the layout of a Web-based publication on the left-side of the GUI, and a print-media based publication on the right-side of the GUI;

[0105] FIG. 2E2 is a schematic representation of the split-screen GUI associated with the computer-based publishing software program of the present invention operated in its UPN/URL data linking mode, showing the layout of a Web-based publication on the left-side of the GUI, and a print-media based publication on the right-side of the GUI;

[0106] FIG. 2E3 is a schematic representation of an exemplary UPN/URL data link table generated during the UPN/URL data link generation mode of operation, and subsequently transported to the UPN/URL database management subsystem of the present invention;

[0107] FIG. 2E4 is a schematic representation of the GUI of an exemplary computer operating system (OS), on which the OS-based UPN/URL data linking function of the present invention is schematically depicted in its compositional/editorial mode, showing a GUI-based window associated with a first arbitrary content-creating application (e.g. Netscape Navigator browsing program), a GUI-based window associated with a second arbitrary content-creating application (e.g., Adobe® Illustrator graphics program), and the UPN/URL data linking GUI enabling the author to create UPN/URL data links between Web documents and UPN-encoded print-documents;

[0108] FIG. 2E5 is a schematic representation of the GUI of an exemplary computer operating system (OS), on which the OS-based UPN/URL data linking function of the present invention is schematically depicted in its UPN/URL Data Linking Mode, showing a GUI-based window associated with a first arbitrary content-creating application (e.g. Netscape Navigator browsing program), a GUI-based window associated with a second arbitrary content-creating application (e.g. Adobe® Illustrator graphics program), and the UPN/URL data linking GUI enabling the author to create UPN/URL data links between Web documents and UPN-encoded print-documents;

[0109] FIG. 3A1 is a graphical representation of a first illustrative embodiment of the client computer system of the present invention, designed for use in desktop environments at home, work and play;

[0110] FIG. 3A2 is a graphical representation of a second illustrative embodiment of the client computer system of the present invention realized in the form of a bar code driven multi-media kiosk, designed for use as a "virtual or Cyber sales agent" in retail shopping environments, such as department stores, supermarkets, superstores, retail outlets and the like;

[0111] FIG. 3A3 is a graphical representation of a third illustrative embodiment of the client computer system of the present invention realized in the form of a bar code driven multi-media kiosk, designed for use as a "Cyber sales agent" in retail shopping environments such as department stores, supermarkets, superstores, retail outlets and the like, and shown as having an integrated "cord-connected" type laser scanning bar code symbol reader disposed overhead its LCD touch-screen panel, a telephone handset for carrying out telephone calls, and a credit card transaction terminal for conducting consumer purchase transactions and other forms of electronic commerce while using the consumer product information finding system of the present invention;

[0112] FIG. 3A3' is a graphical representation of the bar code driven multi-media kiosk shown in FIG. 3A3, wherein

the laser scanning projection-type bar code symbol reader is removed from its support stand, by pulling its connector cable out of its cable take-up unit, and used to read a bar code symbol on product located a relatively short distance away from the kiosk;

[0113] **FIG. 3A4** is a graphical representation of a fourth illustrative embodiment of the client computer system of the present invention realized in the form of a bar code driven multi-media kiosk, designed for use as a "Cyber sales agent" in retail shopping environments such as department stores, supermarkets, superstores, retail outlets and the like, and shown as having an integrated "cordless" type laser scanning bar code symbol reader disposed overhead its LCD touch-screen panel, a telephone handset for carrying out telephone calls, and a credit card transaction terminal for conducting consumer purchase transactions and other forms of electronic commerce while using the consumer product information finding system of the present invention;

[0114] **FIG. 3A4'** is a graphical representation of the bar code driven multi-media kiosk shown in **FIG. 3A4**, wherein the laser scanning projection-type bar code symbol reader is removed from its support stand and used to read a bar code symbol on a product located a relatively short distance away from the kiosk;

[0115] **FIG. 3A5** is a graphical representation of a fifth illustrative embodiment of the client computer system of the present invention realized in the form of a consumer product information access terminal, designed for use as a "sales agent's tool" at a point-of-sale (POS) station in retail shopping environments, wherein the information access terminal has a bar code symbol reader integrated with the POS station for reading the UPC numbers on consumer products being offered for sale in the store, and also a LCD screen capable of being mounted in various viewing positions for displaying consumer product-related information accessed from a centralized database interconnected to the Internet;

[0116] **FIG. 3A6** is a graphical representation of a sixth illustrative embodiment of the client computer system of the present invention realized in the form of a bar code driven multi-media kiosk, which is completely transportable within the store by the hand of a shopper for shopping convenience in retail environments such as department stores, supermarkets, superstores, retail outlets and the like;

[0117] **FIG. 3A7** is a graphical representation of a seventh illustrative embodiment of the client computer system of the present invention realized in the form of a bar code driven multi-media kiosk, mounted upon a shopping cart or other vehicle for shopping convenience in retail environments such as department stores, supermarkets, superstores, retail outlets and the like;

[0118] **FIG. 3A8** is a schematic representation of another embodiment of the transportable bar code driven product information access terminal of the present invention, realized using a Newton MessagePad **130** equipped with Nethopper http client software and a Motorola RF modem PCMCIA card, for wireless access to the Internet;

[0119] **FIG. 3A9** is a schematic representation of the e-mail enabled consumer product information transport subsystem of the present invention, wherein each bar code driven kiosk located on a local or wide area network within a retail shopping environment is provided with e-mail based consumer product information transport mode of operation which enables a consumer, accessing consumer product information on the WWW within the shopping environment, to (i) capture, save and attach the same to an e-mail envelope which can be automatically addressable to the consumer's home or like e-mail address in response to the consumer presenting a bar coded (or mag-stripe encoded) customer loyalty/courtesy card to be automatically read at the kiosk, (ii) transport the envelope to the consumer's e-mail address, and (iii) transport a copy of each such e-mail transaction to a central e-mail database server for consumer market research and related operations;

[0120] **FIG. 3A10A** is a schematic representation of the illustrative embodiment of the e-mail enabled consumer product information transport subsystem of the present invention comprising a plurality of bar code driven kiosks connected to a retail store based local wide area network, and a retailer/local e-mail server for enabling e-mail based consumer product information transport services on each such kiosk;

[0121] **FIG. 3A10B** is a schematic representation of the IPI finding and serving subsystem of the present invention, wherein a plurality of web/e-mail enabled kiosks are mounted to the shelving system installed within a "brick of mortar" type retail shopping environment;

[0122] **FIG. 3A10C** is an elevated site view of the first web/e-mail enabled kiosk shown in **FIG. 3A10B**; wherein the kiosk is arranged so that the bar code symbol reading device integrated therewith projects a laser scanning field from below the touch-type LCD screen panel thereof;

[0123] **FIG. 3A10D** is an elevated side view of the second web/e-mail enable kiosk shown in **FIG. 3A10B**, wherein the kiosk is arranged so that the bar code symbol reading device integrated therewith projects a laser scanning field from above the touch-type LCK screen panel thereof;

[0124] **FIG. 3A11** is a schematic representation of an exemplary relational database structure maintained within the retailer RDBMS connected to the central e-mail server shown in FIGS. **3A9** and **3A10A**;

[0125] **FIG. 3A12** is a schematic representation of an exemplary relational database structure maintained within the manufacturer RDBMS connected to the central e-mail server shown in FIGS. **3A9** and **3A10A**;

[0126] FIGS. **3A13A** through **3A13C**, taken together, show a high-level flow chart for the first illustrative embodiment of the consumer product information (CPI) transport method of the present invention, setting forth the steps carried out when a consumer accesses consumer information from Web/e-mail enabled bar code driven kiosk within a retail shopping environment, and transports the same to the e-mail address of the consumer at home, work or on the road;

[0127] **FIG. 3A14** is a schematic representation of an exemplary graphical user interface (GUI) displayed on the bar code driven consumer product information kiosk of the present invention when the CPI transport service of FIGS. **3A13A** through **3A13C** is being performed;

[0128] FIGS. **3A15A** through **3A15C**, taken together, show a high-level flow chart for the second illustrative embodiment of the CPI transport method of the present

invention, setting forth the steps carried out when a consumer accesses consumer information from a Web/e-mail enabled bar code driven kiosk within a retail shopping environment, and transports the same to the e-mail address of the consumer at home, work or on the road;

[0129] **FIG. 3A16** is a schematic representation of an exemplary graphical user interface (GUI) displayed on the bar code driven consumer product information kiosk of the present invention when the CPI transport service of FIGS. **3A15A** through **3A15C** is performed;

[0130] **FIG. 3B** is a schematic representation of an exemplary display screen produced by a (graphical user interface) Java GUI-based web browser program running on a client subsystem and providing an on-screen IPD Web-site Find Button (e.g. UPC REQUEST™ Central Website Find Button) for instantly connecting to the IPD Web-site (e.g. UPC REQUEST™ Central Website) and carrying out the consumer product information finding and serving method of the present invention;

[0131] **FIG. 3C** is a schematic representation of an exemplary display screen produced by a Java GUI-based Internet browser or communication program running on a client subsystem and displaying a Netscape-style browser "display framework", served from the IPD Web-site (e.g. UPC REQUEST™ Central Website), and supporting or providing a sponsor frame for sponsor advertisement, a control frame with Check-Box type buttons for activating any mode of the IPI finding and serving subsystem, and an information frame for displaying HTML documents (instructions, forms, and the like) in accordance with the principles of the present invention;

[0132] **FIG. 4A1** is a schematic representation of the relational-type IPI Registrant Database maintained by each IPD Server configured into the system of the illustrative embodiment of the present invention, showing the information fields for storing (i) the information elements representative of the UPN (e.g. UPC data structure, EAN data structure, and/or National Drug Code (NDC) data structure), URLs, trademark(s) (TM_i), Company Name (CNi) and company address, Product Description (PD_i), E-Mail Address (EMA_i) thereof symbolically-linked (i.e. related) for a number of exemplary IPI Registrants listed (i.e. registered) with the IPI Registrant Database maintained by each IPD Server, image files for registered consumer products, and consumer product information request (CPIR) enabling Applets for access by retailers, wholesalers, advertisers, Web publishers, and the like, and insertion within the HTML code of Web documents on various types of Internet information servers hosting WWW sites, as well as EC-enabled WWW-sites, EC-enabled stores and/or on-line e-commerce product catalogues, so that when executed, these CPIR-enabling Applets automatically access a categorized URL menu containing URLs (identified in **FIG. 4A2**) specifying the location of manufacturer-linked information resources on the Internet pertaining to a particular UPN-labeled product;

[0133] **FIG. 4A2** is a schematic representation of the information subfield structure of the URL Information Field of the IPI Database of **FIG. 4A1**, showing the Product Advertisement Information Field, the Product Specification (Description/Operation) Information Field, the Product Update Information Field, the Product Distributor/Reseller/Dealer Information Field, the Product Warranty/Servicing Information Field, the Product Incentive Information Field thereof, the Product Review Information Field, the Related Products Information Field, and Miscellaneous Information Fields detailed hereinafter;

[0134] **FIG. 4B** is a schematic representation of the relational-type Non-IPI Registrant Database maintained by each IPD Server that is configured into the IPI Finding and Serving Subsystem of the illustrative embodiment of the present invention, showing the information fields for storing (i) the information elements representative of the Company Name (CNi) and Company Address, Trademark(s) (TM_i) registered by the associated Company, E-Mail Address (EMA_i) thereof symbolically-linked for a number of exemplary Non-IPI registrants listed within the Non-IPI Registrant Database maintained by each IPD Server, and CPIR enabling Applets for access by retailers, wholesalers, advertisers, Web publishers, and the like, and insertion within the HTML code of Web documents on various types of Internet information servers hosting WWW sites, as well as EC-enabled WWW-sites, EC-enabled stores and/or on-line e-commerce product catalogues, for the purpose described above;

[0135] **FIG. 4C** is a schematic representation of the structure of a relational database management subsystem (RDBMS) used to carry out a best-mode embodiment of the IPI Registrant Database represented in FIGS. **4A1** and **4A2** hereof;

[0136] **FIG. 4C1** is a schematic representation illustrating the information fields of the table entitled "Manufacturer" used in the RDBMS shown in FIGS. **4A1** and **4A2** hereof;

[0137] **FIG. 4C2** is a schematic representation illustrating the information fields of the table entitled "Consumer Product" used in the RDBMS shown in FIGS. **4A1** and **4A2** hereof;

[0138] **FIG. 4C3** is a schematic representation illustrating the information fields of the table entitled "Information Resources on the WWW" used in the RDBMS shown in FIGS. **4A1** and **4A2** hereof;

[0139] **FIG. 4D** is a schematic representation illustrating the information fields of the table entitled "Retailer" used in the RDBMS shown in FIGS. **4A1** and **4A2** hereof;

[0140] FIGS. **4E1** and **4E2**, taken together, provide a high-level flow chart describing the steps involved in the first illustrative method of creating, loading, distributing, embedding, displaying, and executing "server-side" consumer product information request (CPIR) enabling Applets when using the system architecture and servlet-based search and display mechanism schematically depicted in **FIG. 2B11**, enabling consumers to automatically search the RDBMS for consumer product information related to a particular UPN-specified product while visiting EC-enabled stores and other WWW sites without disturbing the point of presence of the consumer;

[0141] **FIG. 4F1** is a schematic representation illustrating the method of FIGS. **4E1** and **4E2** being carried out using certain subcomponents of the system depicted in FIGS. **2-1**, **2-2**, and **2B1**, in particular;

[0142] **FIG. 4F2** is a schematic representation illustrating in greater detail the Applet-embedding step of the method of

FIGS. 4E1 and 4E2, carried out using certain subcomponents of the system depicted in FIGS. 2-1 and 2-2;

[0143] FIGS. 4G1 and 4G2, taken together, provide a high-level flow chart describing the steps involved in the second illustrative method of creating, loading, distributing, embedding, displaying, and executing "client-side" consumer product information request (CPIR) enabling Applets when using the system architecture and Applet/CGI-based search and display mechanism schematically depicted in FIG. 2B2, enabling consumers to automatically search the RDBMS for consumer product information related to a particular UPN-specified product while visiting EC-enabled stores and other WWW sites without disturbing the point of presence of the consumer;

[0144] FIG. 4H1 is a schematic representation illustrating the method of FIGS. 4G1 and 4G2 being carried out using certain subcomponents of the system depicted in FIGS. 2-1, 2-2, and 2B2, in particular;

[0145] FIG. 4H2 is a schematic representation illustrating in greater detail the Applet-embedding step of the method of FIGS. 4G1 and 4G2, carried out using certain subcomponents of the system depicted in FIGS. 2-1 and 2-2;

[0146] FIGS. 4I1 and 4I2, taken together, provide a high-level flow chart describing the steps involved in the second illustrative method of creating, loading, distributing, embedding, displaying, and executing "client-side" consumer product information request (CPIR) enabling Applets when using the system architecture and Applet/socket-based search and display mechanism schematically depicted in FIG. 2B3, enabling consumers to automatically search the RDBMS for consumer product information related to a particular UPN-specified product while visiting EC-enabled stores and other WWW sites without disturbing the point of presence of the consumer;

[0147] FIG. 4J1 is a schematic representation illustrating the method of FIGS. 4I1 and 4I2 being carried out using certain subcomponents of the system depicted in FIGS. 2-1, 2-2, and 2B3, in particular;

[0148] FIG. 4J2 is a schematic representation illustrating in greater detail the Applet-embedding step of the method of FIGS. 4I1 and 4I2, carried out using certain subcomponents of the system depicted in FIGS. 2-1 and 2-2;

[0149] FIGS. 4K1 and 4K2, taken together, provide a high-level flow chart describing the steps involved in the fourth illustrative method of creating, loading, distributing, embedding, displaying, and executing "client-side" consumer product information request (CPIR) enabling Applets when using the system architecture and Applet/RMI-based search and display mechanism schematically depicted in FIG. 2B4, enabling consumers to automatically search the RDBMS for consumer product information related to a particular UPN-specified product while visiting EC-enabled stores and other WWW sites without disturbing the point of presence of the consumer;

[0150] FIG. 4L1 is a schematic representation illustrating the method of FIGS. 4K1 and 4K2 being carried out using certain subcomponents of the system depicted in FIGS. 2-1, 2-2, and 2B4, in particular;

[0151] FIG. 4L2 is a schematic representation illustrating in greater detail the Applet-embedding step of the method of FIGS. 4K1 and 4K2, carried out using certain subcomponents of the system depicted in FIGS. 2-1 and 2-2;

[0152] FIG. 4M1 is a graphical illustration of an Internet browser display screen that might be displayed on a client computer subsystem hereof while browsing the Library of CPIR-Enabling Java Applet (HTML tags) maintained within the UPN/URL RDBMS shown in FIGS. 2-1 through 2A, wherein the user (e.g. retail purchasing agent, product catalog manager, advertising agent, or whomever) is provided with the option of viewing and downloading, for each UPN-specified product in the system, an executable file containing the HTML tag for either a client-side or server-side type CPIR-enabling Java Applet associated therewith;

[0153] FIG. 4M2 is a graphical illustration of an Internet browser display screen that might be displayed on a client computer subsystem hereof while browsing a product-specific page of an on-line business-to-business consumer product catalog, wherein the user (e.g. retail purchasing agent, product catalog manager, advertising agent, or whomever) is provided with the option of viewing and downloading, for each UPN-specified product offered for sale in the on-line catalog, an executable file containing the HTML tag for either a client-side or server-side type CPIR-enabling Java Applet associated with the UPN-specified consumer product;

[0154] FIGS. 4N1 and 4N2 set forth graphical illustrations of Internet browser display screens that might be displayed on a client computer subsystem hereof while browsing a WWW information search engine or directory, such as Yahoo, Lycos, or Excite, looking for information on the WWW related to a particular consumer product, and launching a CPI search enabling GUI in accordance with the principles of the present invention by clicking on the (underlying) HTML tag of either client-side or server-side Applet embedded within the HTML code of the displayed WWW page associated with the search engine/directory;

[0155] FIGS. 4O1 and 4O2 set forth graphical illustrations of Internet browser display screens that might be displayed on a client computer subsystem hereof while shopping/browsing an EC-enabled storefront, looking for information on any consumer product which is carried within the retailer's EC-enabled store; and launching a CPI search enabling GUI in accordance with the principles of the present invention by clicking on the HTML tag of a client-side or server-side Applet embedded within the HTML code of the displayed E-store Web page;

[0156] FIGS. 4P1 and 4P2 set forth graphical illustrations of Internet browser display screens that might be displayed on a client computer subsystem hereof while shopping/browsing at a particular catalog page in an EC-enabled store, considering whether or not to make an on-line purchase of a particular consumer product displayed on the catalog page; and then initiating a UPN-directed CPI search according to the principles of the present invention by clicking on the HTML tag of a UPN-encoded client-side or server-side Applet embedded within the HTML code of the displayed catalog page;

[0157] FIGS. 4Q1 and 4Q2 set forth graphical illustrations of Internet browser display screens that might be displayed on a client computer subsystem hereof while visiting an on-line EC-enabled auction site (e.g. at http://www.ebay-

.com) when considering whether or not to place a bid on a particular consumer product displayed within the auction listings thereof, and then launching a CPI search enabling GUI in accordance with the principles of the present invention by clicking on the HTML tag of a URL-encoded client-side or server-side Applet embedded within the HTML code of the displayed on-line auction Web page;

[0158] FIGS. **4R1** and **4R2** set forth graphical illustrations of Internet browser display screens that might be displayed on a client computer subsystem hereof while shopping/browsing at a particular auction page in the EC-enabled auction site of FIGS. **4Q1** and **4Q2**, considering whether or not to place at bid on a particular product up being auctioned, and then initiating a UPN-directed CPI search according to the principles of the present invention by clicking on the HTML tag of a UPN-encoded client-side or server-side Applet embedded within the HTML code of the displayed auction page;

[0159] FIGS. **4S1** and **4S2** set forth graphical illustrations of Internet browser display screens that might be displayed on a client computer subsystem hereof while browsing a WWW page of any business, organization or ongoing concern, seeing an Internet product or service advertisement of particular interest on the WWW page, and then initiating a UPN-directed CPI search according to the principles of the present invention by clicking on the HTML tag of a UPN-encoded client-side or server-side Applet embedded within the HTML code of the displayed WWW page;

[0160] **FIG. 4S3** sets forth a graphical illustration of an Internet browser display screen that might be displayed on a client computer subsystem hereof in automatic response to a consumer selecting a URL displayed in the "Buy On The Web" URL category displayed on the Internet browser display screen of **FIG. 4S2**, thereby enabling the purchase of the advertised product or service at an EC-enabled store or product catalogue specified by the selected URL;

[0161] FIGS. **4T1** and **4T2** set forth graphical illustrations of Internet browser display screens that might be displayed on a client computer subsystem hereof while a consumer is reviewing the performance chart of a particular consumer product company displayed at a particular on-line electronic trading WWW site (e.g. http://www.etrade.com) considering whether or not to buy, keep or sell securities (e.g. stock or bonds) in this consumer product company, and eventually requests specific information about the company's products by initiating a trademark/company name-directed CPI search according to the principles of the present invention by clicking on the HTML tag of a trademark/company name-encoded CPIR-enabling Applet embedded within the HTML code of the displayed performance chart;

[0162] **FIG. 5A** is a schematic diagram illustrating the high level structure of communication protocol that can be used among the client subsystem C_a, the IPD Server S_b, and the Web-based Document Server S_{WD} **(30)** of the IPI Finding And Serving Subsystem hereof when, from any particular client subsystem, the subsystem is engaged is in Manufacturer/Product Registration Mode of operation, requesting as input a URL which automatically connects the client subsystem to the Web Document Server associated with the Manufacturer/Product Registration Subsystem of the present invention;

[0163] **FIG. 5B** is a schematic diagram illustrating the high level structure of a communication protocol that can be used among the client subsystem C_a, the IPD Server S_b, and the IPI Server S_c of the IPI Finding and Serving Subsystem hereof when the subsystem is in its Manufacturer Website Search Mode of operation, requesting as input a UPN (e.g. UPC or EAN) associated with a manufacturer's product, and providing as output the URL of the home page of the manufacturer's Web-site and automatically displaying the same;

[0164] **FIG. 5C** is a schematic diagram illustrating the high level structure of a communication protocol that can be used among the client subsystem C_a, the IPD Server S_b, and the IPI Server S_c of the IPI Finding And Serving Subsystem hereof when the subsystem is in its UPN-Directed Information Access Mode of operation, requesting as input a UPN associated with the consumer product, and providing as output the set of URL(s) registered with the consumer product identified by the UPN within the database of the system and pointing to HTML-encoded documents containing particular types of product-related information;

[0165] **FIG. 5D** is a schematic diagram illustrating the high level structure of a communication protocol that can be used among the client subsystem C_a, the IPD Server S_b, and the IPI Server S_c of the IPI Finding And Serving Subsystem hereof when the subsystem is in its Trademark-Directed Search Mode of operation, requesting as input a trademark and/or company name, and providing as output the product descriptor(s) and a UPN (or set thereof) related to the trademark within the database of the system and pointing to HTML-encoded documents containing particular types of product-related information;

[0166] **FIG. 5E** is a schematic diagram illustrating the high level structure of a communication protocol that can be used among the client subsystem C_a, the IPD Server S_b, and the IPI Server S_c of the IPI Finding And Serving Subsystem hereof when the subsystem is in its Product-Description Directed Search Mode of operation, requesting as input a product descriptor related to the consumer product on which information is sought and providing as output the trademark, company name and URL(s) related to the product descriptor within the database of the system and pointing to HTML-encoded documents containing particular types of product-related information;

[0167] **FIG. 6A** provides a high level flow chart illustrating the steps involved in carrying out the communication protocol shown in **FIG. 5A** when the IPI Finding and Serving Subsystem is in its Manufacturer/Product Registration Mode of operation;

[0168] **FIG. 6B** provides a high level flow chart illustrating the steps involved in carrying out the communication protocol shown in **FIG. 5B** when the IPI Finding and Serving Subsystem is in its Manufacturer Website Search Mode of operation;

[0169] **FIG. 6C** provides a high level flow chart illustrating the steps involved in carrying out the communication protocol shown in **FIG. 5D** when the IPI Finding and Serving Subsystem is in its UPN-Directed Information Access Mode of operation;

[0170] FIGS. **6D1** through **6D3**, taken together, provide a high level flow chart illustrating the steps involved in carrying out the communication protocol shown in **FIG. 5C**

when the IPI Finding and Serving Subsystem is in its Trademark-Directed Search Mode of operation;

[0171] FIG. 6E1 through 6E3, taken together, provide a high level flow chart illustrating the steps involved in carrying out the communication protocol shown in FIG. 5E when the IPI Finding and Serving Subsystem is in its Product-Description Directed Search Mode of operation;

[0172] FIG. 7 is a schematic representation of an exemplary embodiment of the IPI Finding and Delivery Subsystem of the present invention showing the various constituent subsystems thereof; and

[0173] FIG. 8 is a schematic representation of a portfolio of Web-sites supported and managed by the UPN/URL database management subsystem with the assistance of the manufacturer/product registration subsystem and Web-enabled client subsystems operated by manufacturers and/or their agents in accordance with the information management principles of the present invention.

BEST MODES FOR CARRYING OUT THE INVENTION

[0174] Referring to the accompanying Drawings, like structures and elements shown throughout the figures thereof shall be indicated with like reference numerals.

Overview of the System of the Present Invention

[0175] As illustrated in FIG. 1, the consumer-product information collection, transmission and delivery system of the present invention is generally indicated by reference numeral 1 and comprises an integration of information subsystems, namely: an IPI finding and serving subsystem 2 for allowing consumers to find product related information on the Internet (e.g. WWW) at particular Uniform Resource Locators (URLs), using UPC numbers and/or trademarks and tradenames symbolically-linked or related thereto; a UPC Product-Information Subsystem ("UPC Catalog") 3 for providing retailers with accurate up-to-date product information on numerous consumer-products offered for wholesale to retailers by manufacturers registering their products therewith; a Electronic Trading Information Subsystem 4 for providing trading partners (e.g. a manufacturer and a retailer) to sell and purchase consumer goods by sending and receiving documents (e.g. purchase orders, invoices, advance slip notices, etc.) to consummate purchase and sale transactions using either Value Added Network (VAN) based EDI transmission or Internet (e.g. HTTP, SMTP, etc.) based electronic document communications; a Sales Analysis and Forecasting Information Subsystem 5 for providing retailers with information about what products consumers are currently buying at retail stores or expect to be buying in the near future; Collaborative Replenishment Information Subsystem 6 for determining what products retailers can be buying in order to satisfy consumer demand at any given point in time; a Transportation and Logistics Information Subsystem 7 for providing retailers with information about when ordered products (purchased by retailers at wholesale) will be delivered to the retailer's stores; and Input/Output Port Connecting Subsystems 8 for interconnecting the input and output ports of the above-identified subsystems through the infrastructure of the Internet and various value-added EDI networks of global extent. Notably, unlike prior art supply chain management systems, the consumer-product information collection, transmission and delivery system of the present invention embraces the manufacturers, retailers, and consumers of UPC-encoded products, and not simply the manufacturers and retailers thereof. As will become apparent hereinafter, this important feature of the present invention allows manufacturers and retailers to deliver valuable product related information to the consumers of their products, thereby increasing consumer purchases, consumer satisfaction and consumer loyalty. Prior art supply chain management systems have no way or means of providing such information services to the consumers of UPC-encoded products along the consumer-product supply and demand chain.

[0176] As shown in FIGS. 2-1 and 2-2, the consumer-product information collection, transmission and delivery system of FIG. 1 is realized as an arrangement of system components, namely: a central UPN/URL Database Management Subsystem 9 for storing and serving various types of consumer-product information to retailers, manufacturers and consumers alike (e.g., the name of the product's manufacturer; the Universal Product Code (UPC) or European Article Number (EAN) assigned to the product by the manufacturer; one or more URLs specifying the location of information resources on the Internet at which particular kinds of information relating to the consumer-product can be found; merchandise classification; style number; tradename; information specifying the size, color and other relevant characteristics of the consumer-product, where applicable; ordering criteria; availability and booking dates, etc.); a globally-based (packet-switched) digital telecommunications network (such as the Internet) 10 having an infrastructure including Internet Service Providers (ISPs), Network Service Providers (NSPs), routers, telecommunication lines, channels, etc., for supporting packet-switched type digital data telecommunications using the TCP/IP networking protocol well known in the art; one or more Internet Product Finding Directory (IPD) Servers, each indicated by reference numeral 11 and being connected to the Internet at strategically different locations via the Internet infrastructure 10 and data-synchronized with each other in order that each such Server maintains mirrored a relational-type database structure as represented in FIGS. 4A and 4B; a plurality of Internet Product-Information (IPI) Servers, each indicated by reference numeral 12 and being connected to the Internet via the Internet infrastructure; a plurality of retailer-related electronic-commerce (EC) information servers 12A, each operably connected to the infrastructure of the Internet, and enabling the hosting or one or more EC-enabled stores or EC-enabled on-line catalogues (i.e. EC-enabled WWW sites) owned, operated, managed and/or leased by one or more retailers along the retail supply and demand chain; a plurality of manufacturer-related electronic-commerce (EC) information servers 12B, each operably connected to the infrastructure of the Internet, and enabling the hosting or one or more EC-enabled stores or BC-enabled on-line catalogues (i.e. EC-enabled WWW sites) owned, operated, managed and/or leased by one or more manufacturers along the retail supply and demand chain; a plurality of User (or Client) Computers, each indicated by reference numeral 13, being connected to the Internet via the Internet infrastructure and available to consumers (C_1, C_2, C_3, . . . , C_i); one or more data communication (i.e. EDI) networks 14, comprising data collection nodes 15 and communication links 16, operably

connected to the centralized UPN/URL Database Management Subsystem **9**, each Client Computer **13** available to a Manufacturer (M_1, M_2, M_3, . . . , M_j) and Retailer (R_1, R_2, R_3, . . . , R_k) within the retail supply and demand chain; a Web-based Document Server **30** connected to at least one of the IPD Servers **11** and the Internet infrastructure, for transferring documents and messages to remote Client Computer Systems during the registration of manufacturers and consumer products with the system hereof and periodically updating product-related information with the IPD Servers **11** in an automatic manner; and a Web-based Document Administration Computer **31** connected to the Web-based Document Server **30** by way of a TCP/IP connection **32**, for administrating the registration of manufacturers and products with the system, initiating the transfer of consumer product related information (e.g. menu of URLs) between the remote Client Computer Systems and Web-Based Document Server **30**, transferring such information to the IPD Servers **11**, and maintaining local records of such information transfers and the like. As will become apparent hereinafter, Web-based Document Server **30** and Web-based Document Administration Computer **31** provide a subsystem for (i) managing the process of registering qualified manufacturers and their consumer products and related Web pages (e.g. UPC numbers and URLs), and (ii) updating the product-related information with the IPD Servers **11** in a n automatic manner to ensure accurate links between UPNs and URLs within the UPN/URL Database Management Subsystem. The subsystem comprising the Web-based Document Server **30** and Web-based Document Administration Computer **31** shall be referred to as the Manufacturer/Product Registration Subsystem of the consumer product information finding and delivery subsystem **2** and indicated by reference numeral **33** throughout the figure drawings hereof.

[0177] Preferably, the centralized UPN/URL Database Management Subsystem **9** and at least one of the IPD Servers **11** are located at a secured information storage/processing center **17**, along with a multiprocessor (or mainframe) computer system, information servers, routers, data communication lines, disk storage devices (e.g. RAIDs), tape drives and tape-library system, uninterrupted power supplies (UPS), and other peripheral technology to provide on-line, batch and back-up operations. However, the IPI Servers, the Client Computers and the other IPD Servers (if provided for database mirroring purposes), typically will be located throughout the world, as the distribution of manufacturers, retailers and consumers who are encouraged to use the system is scattered across the Planet.

[0178] In the illustrative embodiment, the Web-based Document Server **30** is a Windows NT Server running WebDox™ Server software from Premenos Corporation of Concord, Calif. The Windows NT Server can be realized using a suitable computer system having a Pentium® or higher CPU, 64 MB of RAM or higher, running (i) Microsoft Windows NT Server 4.0 or higher Operating System software from Microsoft Corporation, (ii) Microsoft Internet Information Server 2.0 or higher from Microsoft Corporation, and (iii) Microsoft SQL Server 6.5 or higher software from Microsoft Corporation. Also, the WebDox™ Server is provided with a dedicated Internet connection (i.e. ISDN or better) to the Internet infrastructure **10**.

[0179] The EDI administration computer **31** is either a Windows 95 or Windows NT Computer system running WebDox Admin™ software from Premenos Corporation of Concord, Calif. The Windows 95 or Windows NT computer system **31** can be realized using a suitable computer system having an Intel 486 or higher CPU, 12 MB of RAM or higher, running Microsoft Windows 95 or Windows NT 4.0 or higher, and having a TCP/IP connection **31** to the WebDox™ Server **30**.

[0180] In order to use the WebDox™ system, each remote Client Computer System **13** includes either a Windows 95 or Windows NT Computer system running WebDox Remote™ software from Premenos Corporation of Concord, Calif. The Windows 95 or Windows NT computer system **13** can be realized using a suitable computer system having an Intel 486 or higher CPU, 16 MB of RAM or higher, and a VGA monitor or better, and running (i) Microsoft Windows 95 or Windows NT 3.51 or higher Operating System (OS) software, and (ii) Microsoft Internet Explorer 3.0 or higher from Microsoft Corporation. Also, the WebDox Remote™ Server is provided with a dial-up Internet connection (i.e. 14,400 bps or better) to the Internet infrastructure. The function of the Web-based Document Server **30**, Web-based Administration System **31** and remote client subsystems **13** running the Premenos® WebDox Remote™ software is to provide a Web-based Document Transport System for automatically transferring information (e.g. UPN/URLs) from manufacturers to the IPD Servers of the system in order to periodically update the same. While the illustrative embodiment of this Web-based Document Transport System has been described in terms of its implementation using the WebDox™ system from Premenos, it is understood that other commercially available electronic document transport systems (e.g. COMMERCE:FORMS™ Electronic Business Forms Package from Sterling Commerce, Inc., http://www.stercomm.com) can be used to carry out this subsystem. The operation of this Web-Based Document Transport System will be described in detail hereinafter with respect to the collection and delivery of consumer product-related information to the IPDs hereof.

[0181] The major subsystem components comprising the consumer-product information collection, transmission and delivery system of the present invention will be described in greater detail below.

[0182] In the illustrative embodiment of the present invention, the UPC Product-Information Subsystem **2** is realized using the UPN/URL Database Management Subsystem **9** and data communication networks **14** shown in FIGS. **2-1** and **2-2**. Preferably, the product procurement services delivered by the UPN/URL Database Management Subsystem **9** are provided by modifying the prior art QRSolutions UPC Catalog currently implemented by QuickResponse Services, Inc., so that this subsystem includes the database structures (i.e. information fields and data elements) of the IPD Database Server **11** which are neither found in nor suggested by the prior art QRSolutions UPC Catalog. The structure and operation of the UPN/URL Database Management Subsystem and IPD Server of the present invention will be described in greater detail hereinafter. The information services supported by the UPC Product-Information Subsystem **3** include those provided by the prior art QRSolutions UPC Catalog, and also a number of additional information services that can be used to carry out Product Registration

within the IPI Finding and Serving Subsystem of the present invention. These additional information services will be described in greater detail hereinafter with reference to **FIG. 2A**.

[0183] The Electronic Trading Information Subsystem **4** is realized using the UPN/URL Database Management Subsystem **9**, Client Computer Systems **13** and data communication networks **14** of the technology platform shown in FIGS. **2-1** and **2-2**. Preferably, the inventory procurement services delivered by the Electronic Trading Subsystem **4** are provided by the prior art QRSolutions Econnect and Electronic Data Interchange Services currently being implemented by QuickResponse Services, Inc.

[0184] Sale Analysis and Forecasting Information Subsystem **5** is realized using information storage/processing center **1**, Client Computer Systems **13**, and the data communication networks **14** of the enabling technology platform shown in FIGS. **2-1** and **2-2**. Preferably, the product inventory management services delivered by the Sale Analysis and Forecasting Information Subsystem **5** are provided by the prior art QRSolutions Sale Analysis and Forecasting Information Services currently being implemented by QuickResponse Services, Inc.

[0185] The Collaborative Replenishment Information Subsystem **4** is realized using information storage/processing center **17**, Client Computer Systems **13** and the data communication networks **114** of the enabling technology platform shown in FIGS. **2-1** and **2-2**. Preferably, the product inventory management services delivered by the Collaborative Replenishment Information Subsystem **6** are provided by the prior art QRSolutions Replenishment Services currently being implemented by QuickResponse Services, Inc.

[0186] The Transportation and Logistics Information Subsystem **7** is realized using information storage/processing center **17**, Client Computer Systems **13**, and the data communication networks **14** of the enabling technology platform shown in FIGS. **2-1** and **2-2**. Preferably, the product distribution management services delivered by the Transportation and Logistics Information Subsystem **7** are provided by the prior art QRSolutions EDI and Logistics Management Services currently being implemented by QuickResponse Services, Inc.

[0187] In the illustrative embodiment of the system of the present invention, each Client Computer Subsystem **13** has a conventional Java GUI-based web browser program (e.g. Netscape, Internet Explorer, Mosaic, etc.) with a plug-in type module, such as CyberFinder™ navigational software by Aladdin Systems, Inc., of Watsonville, Calif., that provides an on-screen graphical icon for a "IPI Web-site Find" function. An exemplary display screen **18** produced by such a Java GUI-based web browser program is set forth in **FIG. 3B**. Alternatively, the URL of the home page of the IPI Web-site can be recorded as a browser "bookmark" for easy recall and access through a conventional Java GUI-based Internet browser. Once at the home page of the IPI Web-site, an Internet user can find product-related information on the Internet in essentially the same way as when using the web browser program of **FIG. 3B**. As shown, the on-screen radio button **19** functions as an "IPI Web-site Find" Button (or Consumer Product Information Button) for instantly connecting the client subsystem to a particular IPI Web-site (i.e. hosted on each mirrored IPD Server) and especially adapted for carrying out the IPI finding and serving method of the present invention. As will be described in greater detail hereinafter, examples of "IPI Web-sites" can include, but are not limited to: (1) one or more mirrored UPC Request Central Web-sites from which consumer product information from all manufacturers is available for access to consumers from predetermined Internet domains; and (2) an UPC Request Retail Web-site, for each retailer, wherein consumer product information associated with only manufacturers of products offered by the retailer is available for access to consumers from predetermined Internet domains within physical retail "brick and mortar" stores and "electronic commerce enabled stores.

[0188] In general, each IPI Web-site can be sponsored by a retail store subscribing to the consumer product information service hereof, or by one or manufacturers and/or service providers. The URL for the home page of any particular IPI Web-site can be selected with marketing considerations in mind, for example, "http://www.ipf.com" or "http://www.upcrequest.com" similar in form with the URLs of other information search-engines and directories currently available on the Internet. Upon selecting the IPI Web-site Find Button **19** (e.g. by a clicking of the mouse thereon shown in **FIGS. 3B and 3C**), the user is automatically connected to the home-page of the IPI Web-site (hosted on each mirrored IPD Server) which, as shown in **FIGS. 3B and 3C**, supports a Netscape-style "framework", within which web pages accessed through the IPI web-site are displayed. An excellent tutorial on "framing" entitled "The Netscape Frames Tutorial™ (2nd edition)" by Charlton D. Rose set forth at the URL: "http://www.newbie.net/frames/", last visited by Applicant on Mar. 26, 1997.

[0189] In general, the HTML-encoded documents served from the IPD Servers **11** hereof to the client subsystems **13** hereof will preferably have a three-field Netscape-style display framework which provides a unique and effective way of satisfactorily addressing the needs of consumers, hosting retailers, manufacturers and the IPI provider(s)/publishers alike. In practice, the Netscape-style browser "framework" can simultaneously accommodate the needs of the consumers using the particular Client Subsystems of the present invention, as well as the needs of the retailers who typically will host client subsystems hereof either (1) physically within their stores, and/or (2) electronically on their WWW sites using Web browser framing techniques as well.

[0190] As shown in **FIG. 3C**, the first (top-most) display field, the sponsor frame **20A**, can be used to display to the consumer, a Web page (e.g. HTML-encoded document) containing a message that the IPI Finding and Serving Subsystem is being delivered to the consumer by the IPI Provider under, for example, the sponsorship of either: (1) the hosting retailer; (2) one or more advertisers posting advertising "banners" in the display frame **20A**; or (3) the consumer himself/herself by paying a subscription fee or the like. Understandably, the method of sponsorship employed will vary from embodiment to embodiment of the present invention. An exemplary message for this display screen might read, for example, as follows:

[0191] "Welcome to the UPC Request™

[0192] Consumer Product-Information Finding and Serving System

[0193] sponsored by THE HOME DEPOT

[0194] for your shopping convenience and pleasure-."The height of the sponsor frame **20A** need only be a small fraction of the consumer's display screen (e.g. ¾ inches) to convey this message to the consumers during use of the IPI Finding and Serving Subsystem of the present invention within the retailer's real (or virtual/electronic) shopping environment.

[0195] As shown in **FIG. 3C**, the second (left-most) display field, the control frame **20B**, is used to display an HTML-encoded document containing a Java GUI-based "control panel"**21** for the consumer product information finding and serving subsystem of the present invention. In the illustrative embodiment, this control panel **21** includes six Check Box type buttons. namely: a first Check Box type button **21A** which, when selected, automatically activates the Manufacturer/Product Registration Mode of the subsystem; a second Check Box type button **21B** which, when selected, automatically activates the Manufacturer Website Search Mode of the subsystem; a third Check Box type button **21C** which, when selected, automatically activates the UPN-Directed Information Access Mode of the subsystem; a fourth Check Box type button **21D** which, when selected, automatically activates the Trademark-Directed Search Mode; a fifth Check Box type button **21E** which, when selected, automatically activates the Product-Description Directed Search Mode of operation of the subsystem; and a sixth Check Box type button **21F** which, when selected, automatically activates the UPC-Encoded-Applet-Download/Distribution Mode of operation of the subsystem. Each of these Check Box type buttons is hot-linked to a particular HTML-encoded document residing on the IPD Server(s) **11** of the subsystem hereof.

[0196] While the IPI Web-site of the illustrative embodiment has a framework characterized by three-display fields, namely, the sponsor frame **20A**, the control frame **20B**, and the information display frame **20C**, it is understood, however, that there may be more or fewer display frames than that shown in **FIG. 3C**. Each frame will act as a separate display screen where variables such as web-pages, scrolling, page colors, etc., are independently controllable.

[0197] As will become apparent hereinafter, one of the primary functions of the client subsystems **13** hereof is to provide UPN-driven consumer product information (CPI) GUIs within both "physical "brick and mortar" retail stores" and "E-commerce" enabled retail stores and product catalogues. Hereinafter, UPN-driven CPI GUIs provided within physical retail shopping environments will be referred to as "physical" or "physically-based" UPN-driven CPI kiosks, whereas UPN-driven CPI GUIs provided within E-commerce enabled retail shopping environments will be referred to as "cyber" UPN-driven CPI kiosks, despite the fact that these devices may provide the substantially the same type of consumer product information services to consumers, retailers and manufacturers along the retail supply and demand chain.

[0198] Physically-based UPN-driven CPI consumer product information kiosks will have great utility in physical retail shopping environments. However, such subsystems will be of little value to consumers browsing the Internet and shopping at EC-enabled WWW sites, unless they are located in "brick and mortar" type retail stores wherein consumers are provided with the option of shopping and conducting e-commerce transactions therein for all or selected items of merchandise offered for sale by the retailer. Moreover, when shopping in any particular retailer's EC-oriented store, however realized, it is also understood that great efforts must be undertaken to ensure that the shopper does not leave the EC-oriented store prior to making a purchase at the checkout page of the EC-oriented WWW site. Requiring, prompting or otherwise encouraging a shopper to link over to the IPD WWW site hereof (e.g. hosted on the IPD information server) for desired consumer product related information oftentimes presents a great risk that the shopper will not return to the EC-oriented store, at which he or she was once visiting, but rather will visit another EC-oriented store to make the product purchase.

[0199] The above limitations of physically-based consumer product information kiosks and the risks associated with consumer behavior while shopping on the Internet are overcome by the UPN-based cyber kiosks of the present invention. The primary function of UPN-based cyber CPI kiosks is to provide consumers with a simple and effective way of and means for producing UPN-driven CPI graphical user interfaces (GUIs) at the consumer's point of presence (POP) which may exist, for example, when: (i) shopping at EC-enabled stores, product catalogs and other types of EC-oriented WWW sites; (ii) reviewing and responding to Internet-based product advertisements (including Web-based discount coupons and the like) published at selected sections of Web-documents served from diverse types of WWW sites hosted on the millions of Internet information servers connected to the infrastructure of the Internet; and/or (iii) encountering a Web-document addressing a particular consumer product under review, analysis or other form of observation where accurate consumer product related information is desired or required by the consumer, whomever they might be. The details of producing UPN-enabled CPI GUIs in both physical and cyber retail environments will be described hereinafter.

[0200] As shown in **FIG. 1**, each synchronized IPD Server **11** is interfaced with an ISP **10A** in a conventional manner. The actual number of IPD Servers **11** used in any particular application will depend on various factors including, for example, user demand, Internet traffic conditions, network router capacity and performance, etc. Each such IPD Server **11** is assigned a static IP address and a common domain name on the Internet according to the Domain Name System (DNS) well known in the art. Data synchronization among such databases can be achieved using conventional data synchronization techniques well known in the art. In addition, a backup and mirroring program can be used to maintain data security. Preferably, the synchronized IPD Servers are maintained by a team of network managers under the supervision of one or more webmasters.

[0201] As shown in FIGS. **2B1** through **2B4**, using presently known technology available for use on the WWW, there are at least four different ways of configuring IPD Server **11** and back-end UPN/URL Database Management Subsystem **9** of the illustrative embodiment. These four different subsystem architectures are schematically depicted in FIGS. **2B1** through **2B4**.

[0202] In the system architectures shown in FIGS. **2B2** through **2B4**, client-side Applets ("Applets"), having their

<APPLET>HTML tags embedded within HTML documents (e.g. using the HTML 3.2 Specification), are executed with Java-enabled browsers on the client-side of the information network. In the system architecture set forth in **FIG. 2B1**, server-side Applets ("Servlets"), having their <SERVLET>HTML tags embedded within HTML documents (e.g. the HTML 3.2 Specification), are executed within Java-enabled Web servers on the server-side of the information network. Collectively, client-side Applets and server-side Applets shall be referred to as "Applets", wherein the major distinction between these two types is based on where the Applet is executed on the network (i.e. client-side or server side).

[0203] In each of these four system architectures, the IPD Server **11** performs a number of basic functions, for example: (1) serving HTML-encoded documents associated with IPD Web-sites (e.g. UPC Request Central WWW site, UPC Request Retail WWW sites, etc.) to client subsystems **13** on the Internet so as to enable the six primary modes of operation of the consumer product information finding and delivery subsystem hereof including, but not limited to, access to consumer product related information stored within the IPI and Non-IPI Registrant Databases on the UPN/URL Database Management Subsystem **9**; as well as (2) serving Libraries of executable files containing "UPN-enabled Java Applet tags" for client-side Applets as well as server-side Applets a/k/a "Servlets", so as to enable retailers, manufacturers, advertisers, et al to download the executable "Applet tag containing" file to client subsystems.

[0204] According the first system architecture shown in **FIG. 2B1**, the UPN/URL Database management Subsystem **9** is realized by a SQL-based RDBMS server **9**, whereas the IPD server **11** is realized by a Java Web Server **11'**, provided with Java servlet support, and operably connected to the RDBMS server **9** by way of high-speed digital transmission link known in the art. During system operation, the Java Web Server **11'** serves to a Java-enabled client subsystem **13**, an HTML-encoded document containing a servlet HTML tag <SERVLET> which, upon selection by a single mouse-clicking operation by the consumer, sends an http request to the Java Web Server **11'**, invoking a prespecified UPN-encoded servlet stored therewithin, causing the CPIR-enabling servlet to execute on the server-side of the network. This causes the servlet to call and run certain predefined Java methods which carry out a UPN-specified CPI search on the RDBMS server **9** and return the search results to the client subsystem **13** for display within a predetermined GUI generated therewithin. Using this system architecture, each UPN-encoded servlet executed within the Java Web Server **11'** will contain information relating to (1) the UPN-specified consumer product on which product information is to be searched for within the RDBMS server **9**, (2) licensing information relating to whom the CPIR-enabling servlet has been licensed (although this architecture does not enable easy enforcement of the granted license as the servlet is executed on the server side of the network.

[0205] According to the second system architecture shown in **FIG. 2B2**, the UPN/URL Database management subsystem **9** is realized by a SQL-based RDBMS server **9**, whereas the IPD server **11** is realized by a Java Web Server **11"**, providing Java Applet support and being operably connected to the RDBMS Server **9** by a high-speed digital data transmission link known in the art. During system operation, the Java Web Server **11"** serves to the Java-enabled client subsystem **13**, an HTML-encoded document containing a "UPN-encoded" Applet HTML tag <APPLET> which, upon selection by a single mouse-clicking operation by the consumer, causes the CPIR-enabling Applet to execute on the client-side of the network, sending an http request to the Java Web Server **11"**, invoking a prespecified Common Gateway Interface (CGI) stored within the Java Web Server **11"**. This causes the Applet to call and CGI to run certain predefined methods for carrying out a UPN-specified CPI search on the RDBMS server **9** and returning the search results to the client subsystem **13** for display within a predetermined GUI prespecified within the Applet. Using this system architecture, each UPN-encoded Applet executed within the Java browser of the client machine **13** will contain information relating to (1) the UPN-specified consumer product on which product information is to be searched for within the RDBMS serverm**9**, (2) licensing information relating to whom the client-side Applet has been licensed and by whom the Applet may be served within the terms of the licensing program, etc.

[0206] According to the third system architecture shown in **FIG. 2B3**, the UPN/URL Database Management Subsystem **9** is realized by a SQL-based RDBMS server **9**, whereas the IPD server **11** is realized by a Java Web Server **11'"**, providing client-side Applet support, and being operably connected to the RDBMS server **9** by way of a high-speed digital data transmission link known in the art. During system operation, the Java Web Server **11'"** serves to the Java-enabled client subsystem **13**, an HTML-encoded document containing a UPN-encoded Applet HTML tag <APPLET> which, upon selection by a single-mouse clicking operation by the consumer, causes the Applet to execute on the client-side of the network, creating a "socket-type" connection at lower (TCP/IP) communication layers between the client subsystem **13** and Java Web Server **11'"**, enabling the Java Web Server **11'"** to run certain predefined Java methods for carrying out a UPN-specified CPI search on the RDBMS server **9**, and returning the search results to the client subsystem **13** for display within a GUI prespecified within the Applet. Using this system architecture, each UPN-encoded Applet executed within the Java client subsystem **13** will be created to contain information relating to (i) the UPN-specified consumer product on which product information is to be searched for within the RDBMS server **9**, (ii) licensing information relating to whom the client-side Applet has been licensed and by whom the Applet may be served within the terms of the licensing program, etc.

[0207] According to the fourth system architecture shown in **FIG. 2B4**, the UPN/URL Database Management Subsystem **9** is realized by a SQL-based RDBMS server **9**, whereas the IPD Server **11** is realized by a Java Web Server **11""**, supporting client-side Applet execution and being operably connected to a high-speed digital data communication link well known in the art. During system operation the Java Web Server **11""** serves to the Java-enabled client subsystem **13**, an HTML-encoded document containing a Applet HTML tag <APPLET> which, upon selection by a single mouse-clicking operation by the consumer, causes the CPIR-enabling Applet to execute within the Java-enabled client **13** on the client-side of the network, calling a Remote Invocation Method to carry out a prespecified CPI search on the RDBMS server **9** and returning the search results to the client subsystem **13** for display within a predetermined GUI

prespecified by the Applet. Using this system architecture, each UPN-encoded Applet executing within the Java enabled client **13** will contain information relating to (1) the UPN-specified consumer product on which product information is to be searched for within the RDBMS server, (2) licensing information relating to whom the server-side Applet has been licensed and by whom the Applet may be served within the terms of the licensing program, etc.

[0208] In the first illustrative embodiment shown in **FIG. 2B1**, Java (enabled) Web Server **11'** can be realized by, for example, the Origin 200 Server or the O_2 Desktop Workstation from Silicon Graphics, Inc, a high-end SUN information server from Sun Microsystems, Inc., or any other suitable computing machine, running: (1) JDBC Interface software for providing a uniform access to a wide range of relational databases on RDBMS server **9** (if necessary in a particular application of the system hereof) and providing a common base on which higher level tools and interfaces can be built; and (2) a servlet-enabled Web (http) server software program such as, the Java Web Server (JWS) 1.0 or later from JavaSoft, division of Sun Microsystems, Inc., or the JigSaw Web Server from the World Wide Web Consortium, each proving native Java support, or alternatively, the Fastrak™ Web (http) server from Netscape Communications, Inc., the Internet Information Server (IIS) from the MicroSoft Corporation, the Apache HTTP Server from The Apache Software Foundation at http://www.apache.org, or any other http server capable of transporting HTML-encoded documents, in conjunction with the Java Servlet Developer's Kit from JavaSoft, or the Servlet Express Tool from IBM Research Labs in Haifa, Israel, for managing servlets on Web servers lacking native Java support. In order to develop servlets, the Java Web Server **11'** should also be equipped with the following software tools: the Sun Java Developers Kit 1.1.x from Sun Microsystems, Inc.; and the Java Servlets Development Kit (JDSK) from Sun Microsystems, Inc., or a Java Development Environment that supports JDK 1.1.x, such as VisualAge for Java by IBM, MicroSoft's Visual J++, or the like. Optionally, the Java Web Server **11'** may also include Web-site development software (e.g. based on the HTML 3.2 or 4.0 Specification) for creating and maintaining the IPI Web-sites of the present invention, although such tools will be typically run on client subsystem **13** for practical reasons.

[0209] In the illustrative embodiments of FIGS. **2B2** through **2B4**, Java Web Servers **11"** through **11""** can be realized by, for example, the Origin 200 Server or the O_2 Desktop Workstation from Silicon Graphics, Inc, a high-end SUN information server from Sun Microsystems, Inc., or any other suitable computing machine, running: (1) JDBC Interface software for providing a uniform access to a wide range of relational databases on RDBMS server **9** (if necessary in a particular application of the system hereof) and providing a common base on which higher level tools and interfaces can be built; (2) a Web (http) server such as the Java Web Server (JWS) from JavaSoft, the JigSaw Web Server from the World Wide Web Consortium, the Internet Information Server (IIS) from the MicroSoft Corporation, the Apache HTTP Server from the Apache Software Foundation, or other Java-enabled Web server capable of transporting HTML encoded documents; (3) the Sun Java Developers Kit, from Sun Microsystems, Inc., for developing client-side Applets; and (4) optionally, Website development software (e.g. based on the HTML 3.2 or 4.0 Specification) for creating and maintaining the IPI Web-sites hereof, although such tools will typically run on client subsystems **13** for practical reasons. Notably, when using the Microsoft IIS, one can use a Java Development Environment that supports JDK 1.1.x, such as VisualAge for Java by IBM, MicroSoft's Visual J++, and the like. Also, Java Web Server **11"** must provide support for running CGI scripts written in Java, PERL or other suitable scripting language known in the art.

[0210] In the illustrative embodiments shown in FIGS. **2B1** through **2B4**, each SQL-based RDBMS Server **9** can be realized by, for example, the Origin 200 Server from Silicon Graphics, Inc., the O_2 Desktop Workstation from Silicon Graphics, Inc., a ULTRA™ information server from Sun Microsystems, Inc., or any other suitable computing machine, running a RDBMS software program such as ORACLE 8.0 from Oracle Corporation, Sybase SQL from Sybase, Inc., Access 98 from Microsoft, or other database development program based on a database programming language such as the SQL Language, the Sybase language, or any other suitable database language enabling database programming and connectivity over the Internet.

[0211] In principle, there can be millions of IPI Servers **12** within the system hereof, each enabled to serve Web-based documents containing consumer product related information. Notably, each such IPI Server **12** can be realized by, for example, the Origin 200 Server from Silicon Graphics, Inc, the O_2 Desktop Workstation from Silicon Graphics, Inc., the ULTRA™ information server from Sun Microsystems, Inc., or any other computing machine (e.g. desktop, palmtop, laptop, etc.) running an operating system (e.g. UNIX, LINUX) Macintosh, MS Windows, NT, etc.) capable of performing the functions of a n Internet (http) information server in a client-server distributed object computing environment. As shown in **FIG. 2-1** and **2-2**, each IPI Server **12** is interfaced with an ISP **10A** in a conventional manner. Each such IPI Server **12** is assigned a static IP address and a unique domain name on the Internet. Each IPI Server **12** is also provided with (i) Web-site development software for creating HTML-encoded multi-media pages for Web-site development, (ii) a dynamic web-site auction hosting software solution, such as, AuctionNow 4.2 from OpenSite, Inc. at http://www.opensite.com; and (iii) Web-site server software for supporting HTTP and serving HTML, XML and other document formats used to construct hypermedia-type Web-sites containing product related information of a multimedia nature. Such Web-sites can be expressed in HTML, XML, SGML and/or VRML or any other suitable language which allows for Web-site construction and Web-site connectivity. Web-site management software can be used to maintain correct hyper-links for any particular Web-site. Preferably, the IPI Servers **12** are maintained by a team of network managers under supervision of one or more webmasters.

[0212] Each retailer-related electronic-commerce (EC) information server **12A** indicated in FIGS. **2-1** and **2-2** is operably connected to the infrastructure of the Internet. In general, each retailer-related information server **12A** can be realized by, for example, the Origin 200 Server or O_2 Desktop Workstation from Silicon Graphics, Inc., a high-end information server from Sun Microsystems. Inc., or any other computing machine that can perform the function of a Server in a web-based, client-server type computer system

architecture of the illustrative embodiment. As shown in FIGS. **2-1** and **2-2**, each retailer-related EC-enabled information server **12A** is interfaced with an ISP **10A** in a conventional manner, and is assigned a static IP address and a unique domain name on the Internet. Each retailer-related EC-enabled information server **12A** is also provided with: (i) Java-enabled WWW (http) server software, such as Netscape Communications FastTrak Information Server software, for supporting http, ftp, XML/ICE and other Internet protocols, and serving HTML and XML formatted documents (i.e. pages) associated with Web-sites containing product related information of a multi-media nature; (ii) an advanced EC-enabled product merchandising software solution, such as the Host and Merchant (or Enfinity) Intershop 4 E-Commerce Server Solution from Intershop Communications, Inc., of San Francisco, Calif., and/or catalogMANAGER® and catalogMAKER® software programs from RealEDI, Inc. of Sherman Oaks, Calif., for building, managing and operating all aspects of e-commerce WWW sites, whether implementing on-line merchandising solutions for retailers and manufacturers, creating business-to-business and business-to-consumer product catalogs; (iii) an Internet Advertisement Management Software Solution, such as OPEN ADSTREAM™ Internet AD management software solution by REAL-MEDIA, Inc. of New York, N.Y.), for managing all aspects of Internet advertising on Internet information servers; (iv) a dynamic web-site auction hosting software solution, such as, AuctionNow 4.2 from OpenSite, Inc. at http://www.opensite.com; and optionally (v) Web-site development software for enabling the creation of HTML-encoded multi-media pages and the like for the EC-enabled Web-site development. Such EC-enabled Web-sites can be expressed in HTML, XML and/or VRML or any other suitable language which allows for Web-site construction and Web-site connectivity. Web-site management software can be used to maintain correct hyper-links for any particular Web-site. Preferably, each EC-enabled retailer-related server **12A** is maintained by a team of network managers under supervision of one or more webmasters. The primary function of each retailer-related EC information server **12A** is to enable the hosting of one or more EC-enabled stores or EC-enabled on-line catalogues (i.e. WWW sites) owned, operated, managed and/or leased by one or more retailers, (and optionally wholesalers and manufacturers as well) along the retail supply and demand chain. The use of the Intershop 4 Hosting and Merchant E-commerce software solution enables sellers to design and build dynamic environments for buyers and sellers by enabling sellers (i.e. vendors) to: (1) create a unique look and feel for their e-commerce sites using a Web browser; (2) fully customize their e-commerce sites to maximize the buyers experience, using an import/export function for easily importing existing product databases and site design directly into the Intershop; (3) build detailed profiles of buyers and present them with products that match these profiles, creating a personalized shopping experience; and (4) offer complementary products for sale based on current selections, thereby raising the overall value of each e-commerce transaction carried out. Also, the back-office portion of the Intershop 4 E-commerce Solution is intuitively organized to make it easy for sellers to manage their on-line business through a Web browser.

[0213] Each manufacturer-related electronic-commerce (EC) information server **12B** indicated in FIGS. **2-1** and **2-2** is operably connected to the infrastructure of the Internet. In general, each manufacturer-related EC information server **12B** can be realized by, for example, the Origin 200 Server from Silicon Graphics, Inc., the O_2 Desktop Workstation from Silicon Graphics, Inc., the ULTRA™ information server from Sun Microsystems, Inc., or any other computing machine that can perform the function of a http server in a client-server distributed object computing environment. As shown in FIGS. **2-1** and **2-2**, each manufacturer-related EC-enabled information server **12B** is interfaced with an ISP **10A** in a conventional manner, and is assigned a static IP address and a unique domain name on the Internet. Each manufacturer-related EC-enabled information server **12B** is also provided with: (i) Java-enabled WWW (http) server software, such as Netscape Communications FastTrak Information Server software, for supporting http, ftp, and other Internet protocols, and serving HTML and XML formatted documents (i.e. pages) associated with Web-sites containing product related information of a multi-media nature; (ii) an advanced EC-enabled product merchandising software solution, such as the Host and Merchant Intershop 4 E-Commerce Server Solution from Intershop Communications, Inc., of San Francisco, Calif., and/or catalogMANAGER® and catalogMAKER® software programs from RealEDI, Inc. of Sherman Oaks, Calif., for building, managing and operating all aspects of e-commerce WWW sites, whether implementing on-line merchandising solutions for retailers and manufacturers, or creating business-to-business and business-to-consumer product catalogs; (iii) an Internet Advertisement Management Software Solution, such as OPEN ADSTREAM™ Internet AD management software solution by REAL-MEDIA, Inc. of New York, N.Y.), for managing all aspects of Internet advertising on Internet information servers; (iv) a dynamic web-site auction hosting software solution, such as, AuctionNow 4.2 from OpenSite, Inc. at http://www.opensite.com; and optionally (v) Web-site development software for enabling the creation of HTML-encoded multi-media pages and the like for the EC-enabled Web-site development. Such EC-enabled Web-sites can be expressed in HTML, XML, SGML and/or VRML or any other suitable language which allows for Web-site construction and Web-site connectivity. Web-site management software can be used to maintain correct hyper-links for any particular Web-site. Preferably, each EC-enabled manufacturer-related server **12B** is maintained by a team of network managers under supervision of one or more webmasters. The primary function of each manufacturer-related EC information server **12B** is to enable the hosting or one or more EC-enabled stores or EC-enabled on-line catalogues (i.e. WWW sites) owned, operated, managed and/or leased by one or more manufacturers, (and optionally wholesalers and retailers as well) along the retail supply and demand chain.

[0214] Each Client Computer Subsystem (hereinafter "client subsystem") **13** can be realized by any computing system employing operating system (OS) software (e.g. Macintosh, Windows 95, Windows NT, Unix, etc.) which supports a Java-enabled Internet browser program (e.g. Netscape's Navigator, MicroSoft's Explorer, NCSC's Mosaic, etc.). The operating system should also include: (1) Internet networking software that supports the TCP/IP networking protocol (required by HTTP, FTP and the like) and provides a JAVA GUI-based Web browser interface; and, in the case of client computer machines **13** that are used by manufacturers and retailers in their "back office" operations, (2) Electronic Data Interchange (EDI) networking software that

supports all versions of EDI between two or more client subsystems over the VAN-based or Web-based EDI networks illustrated in FIGS. 2-1 and 2-2. Alternatively, client subsystems may also be realized by any of the following systems: (i) a Newton MessagePad 130 (running the Newton 2.0 Operating System and NetHopper™ Internet Software and equipped with a Motorola RF PCMCIA modem card); (ii) a Pippin™ computer system from Apple Computer, Inc.; (iii) a PalmPilot VII wireless Internet-enabled palmtop computing device by 3COM, Inc.; (iv) a network computer (NC) that supports the Java™ programming language and Java applets expressed therewith; (v) a Sony® WebTV Internet Terminal (supported by the WebTV Service provided by WebTV Network, Inc.); or the like. As shown in **FIG. 1**, each Client Computer is interfaced with an ISP **10A** in a conventional manner. Each such client subsystem may be assigned a static IP address and a unique domain name on the Internet, or one may be dynamically assigned thereto by way of its ISP depending on its connectivity, and set of assigned functions within the consumer product information network of the present invention. Optionally, any client subsystem may include Web-site (http) server software serving Web documents of various formats (HTML, XML, SGML or the like) from one or more hypermedia-type Web-sites in a manner well known in the art.

[0215] Typically, each client subsystem **13** will be maintained by either present or future manufacturers, retailers and/or consumers of products, about which information can be found on the Internet. As shown in **FIG. 3A1**, any client subsystem of the present invention may be realized as a desktop computer workstation comprising: a processor and memory **19**; a visual display monitor **20**; a keyboard **21**; a JAVA GUI mouse **22**; and a bar code symbol reader **23** for reading UPC, UPC/EAN and other types of bar code symbols printed on consumer products, brochures, documents, and the like.

[0216] As shown in **FIG. 3A2**, any Client Computer **13** may also be realized in the form of a Web-based (wired or wireless) multi-media kiosk, designed for use as a "Cyber sales agent" within retail shopping environments. As shown in **FIG. 3A2**, the Web-based kiosk of the present invention may comprise: a floor, wall or ceiling supported housing **25**; an omnidirectional laser bar code symbol reader (e.g. Metrologic MS 6720 Laser Scanner) **26** for reading UPC (and other types of) symbols printed on products, brochures, documents and the like; an active-matrix LCD-type visual display screen **27** for viewing product related information automatically displayed thereon in response to the entry of the UPC numbers scanned into the UPC Number Entry Window **21D** below the IPI Finder button **21A** of Control Strip **20B** displayed on the client subsystem, as shown in **FIG. 3C**; a touch-screen type keyboard and pointing device **28** for clicking on anchored links on Web pages, entering information into client subsystem during its use; audio-speakers **29A** for supporting multi-media Web-sites that may be visited when using the client subsystem; a color or black/white printer for printer **29B** for printing out Web pages under consumer command during an information finding session using the system; and also, one or more floppy-disc (or otherwise removable) drive units **29C**, accessible to the consumer for recording promotional and trial versions of information-based consumer products (e.g. video and audio recordings, computer software products, and the like) on removable information storage media (e.g. 1.44 MB floppy discs, 100 MB Zip® floppy discs, 1GB Jazz® floppy discs, etc.) supplied by either the retailer or consumer. Optionally, the kiosk can be provided with a stereoscopic micropolarizing LCD panel from VRex, Inc. of Elmsford, N.Y. so that micropolarized spatially-multiplexed images (SMIs) of 3-D objects represented with VRML-encoded Web pages can be stereoscopically perceived by consumers when viewed through either an electrically-passive polarizing visor structure supported from the housing of the kiosk, or a pair of polarizing eyeglasses tethered to the kiosk housing and donned by the consumer. Notably, by virtue of its compact size and low power requirements, this Web-based kiosk can be easily located in supermarkets, department stores, superstores, home-centers, discount retail outlets, or any other public location where consumer-products are being sold, offered for sale, and/or serviced.

[0217] As shown in **FIG. 3A3**, any Client Computer **13** within the system hereof may be realized in the form of the Web-based multi-media kiosk **34**, also designed for use as a "virtual sales agent" within retail shopping environments. As shown, the Web-based kiosk **34** comprises: an ultra-compact housing **35** capable of being supported upon a pair of support rods (**35A**), a vertical support surface (e.g. wall), a horizontal support surface (e.g. countertop), or supported from a ceiling or pedestal; an omnidirectional laser bar code symbol reader (e.g. Metrologic MS 6720 Laser Scanner) **36**, modified with handle **36A**, for reading UPC (and other types of) symbols printed on products, brochures, documents and the like; an active-matrix LCD-type visual display screen **37** for viewing product related information automatically displayed thereon in response to the entry of the UPC numbers scanned into the UPC Number Entry Window **21D** displayed on the client subsystem; a touch-screen type keyboard and pointing device **38** for clicking on anchored links on Web pages, entering information into client subsystem during its use; audio-speakers **39A** for supporting multi-media Web-site that may be visited when using the client subsystem; a color or black/white printer for printer **39B** for printing out Web pages under consumer command during an information finding session using the system; a scanner support stand **40** with Java GUIde flanges **41A** and **41B**, for Java GUIdably receiving and supporting the scanner **36** as shown in FIG. **3A3**; a recoilable scanner cable **42**, dispensed from cable cartridge **43** and Java GUIded through hole **44** in a scanner support bridge **40**; a telephone handset **45** and associated communication apparatus for making telephone calls over a public telecommunications switching network (PSTN) independent of the operation of the Web-browser of the kiosk; and a mag-stripe card reader **46** and associated credit transaction terminal for automatically dialing up consumer credit and like databases over the PSTN (or Internet) upon scanning mag-stripe card **47** through reader **46**. Optionally, the kiosk may also include one or more floppy-disc (or otherwise removable) drive units (not shown) accessible to the consumer for recording promotional and trial versions of information-based consumer products (e.g. video an audio recordings, computer software products, and the like) on removable information storage media (e.g. 1.44 MB floppy discs, 100 MB Zip(D floppy discs, 1 GB Jazz® floppy discs, etc.) supplied by either the retailer or a consumer. Also, the kiosk can be provided with a stereoscopic micropolarizing LCD panel from VRex, Inc. of Elmsford, N.Y. so that micropolarized spatially-multiplexed images (SMIs) of 3-D objects represented with VRML-encoded Web pages can be

stereoscopically perceived by consumers when viewed through either an electrically-passive polarizing visor structure supported from the housing of the kiosk, or a pair of polarizing eyeglasses tethered to the kiosk housing and donned by the consumer. Notably, by virtue of its compact size and low power requirements, this Web-based kiosk can be easily located in supermarkets, department stores, superstores, home-centers, discount retail outlets, or any other public location where consumer-products are being sold, offered for sale, and/or serviced.

[0218] As shown in **FIG. 3A3**, the bar code symbol reader is supported within its support stand/bridge **40**. In this configuration, the laser scanning field of the reader is projected downwardly upon the surface of the LCD touch screen display panel. By virtue of the angle of tilt of the display panel **37** relative to the ground surface of the retail store, and the projection angle of the laser scanning field relative to the display panel surface, the consumer will be able to easy read the bar code symbol on most consumer products by simply presenting the bar code symbol to the scanning window. In the event that the product is too large to lift from the floor to the scanning window, the consumer can simply remove the bar code symbol reader **36** from its support stand **40**, as shown in **FIG. 3A3'**, by pulling cord **42** out of its take-up compartment **43** so that the reader is positioned to read the bar code symbol **49** on the retail consumer product **48**. When symbol scanning is completed, the bar code symbol reader is lifted back into its stand support position, between support Java GUIdes **41A** and **41B**, while the cord **42** is automatically recoiled back into storage compartment **43**, as shown in **FIG. 3A3**. While the consumer uses the kiosk to scan UPC (or UPC/EAN) symbols on products, to find, access and display consumer product-related information on the display panel **37**, he or she may choose or need to use telephone **45** to speak with a manufacturer's representative and engage in electronic commerce, and/or use the magstripe card reader **46** to read magstripe cards (e.g. credit cards) to pay for consumer purchases made over the Internet using the kiosk of the present invention.

[0219] As shown in **FIG. 3A4**, the Web-enabled kiosk of **FIG. 3A3** is modified to include a bar code symbol reader having a "cordless-type" scanner interface, thereby eliminating the need for the communication/scanner cable **42** shown in **FIG. 3A3**. RF-based wireless interfaces, as disclosed in US Letters Patents and Published International Patent Applications, incorporated herein by reference, can be used to realize this cordless-type scanner interface arranged between the bar code symbol reader **36** and the Web-enabled access terminal integrated within the information kiosk. In all other respects, the kiosk shown in FIGS. **3A4** and **3A4'** is similar to the kiosk shown in FIGS. **3A3** and **3A3'** and described above.

[0220] In **FIG. 3A5**, a fifth illustrative embodiment of the client computer system hereof is realized in the form of a consumer product information access terminal integrated within a point-of-sale (POS) station in retail shopping environments. While this embodiment of the client computer system hereof is particularly adapted for use by sales clerks at POS stations, as well as by store employees behind retail information/service counters, it may also be used by consumers and shoppers alike provided the necessary accommodations are made as described hereinbelow.

[0221] As shown in **FIG. 3A5**, a client subsystem **13** hereof is realized as consumer product information access terminal **60** comprising: a POS station **61** having a cash register computer **61A** and keyboard **61B**, and a price/UPC Database **61C** containing price and UPC number information tables; a Web-enabled computer terminal **62** connecting the POS station **61** to the Internet infrastructure **10** through an ISP **10A**; a bar code symbol reader **63** connected to the POS station **61**; a **15'** diagonal active-matrix LCD panel **64**, operably connected to the output of the Web-enabled computer system **62** and the output of the cash register computer **61A**, and having a swivel-base **65** that allows the LCD panel to be oriented in various viewing positions for displaying consumer product-related information accessed from the IPI Registrant Database shown in FIGS. **4A1** and **4A2**, as well as price information accessed from the price/UPC database **61C**. The advantage of this client computer subsystem is that it enables a retail sales clerk to check out customer purchases in a conventional manner, and conveniently access the IPI Finding and Serving Subsystem when check-out business is relatively slow, to answer any questions that consumers may have regarding a particular product in the retail store. This system will be ideal in retail environments having a high level of customer service and large retail service staff. In such instances, the IPI Finding and Serving Subsystem hereof empowers retail sales clerks, at the POS counters as well as customer service counters, by enabling them to quickly access any item of product related information linked to products in their stores by manufacturers and their agents.

[0222] As taught in the Objects and Summary of the Present Invention set forth hereinabove, the client computer system of the present invention **13** may also be realized in the form of a transportable bar code driven multi-media kiosk which is completely transportable within the store by hand for the convenience of consumers in retail shopping environments as shown in **FIG. 3A6**. As taught hereinabove, the retail shopping environment may be a department store, supermarket, superstore, retail outlet or the like. Notably, the transportable bar code driven multi-media kiosk shown in **FIG. 3A6** is similar to the bar code driven multi-media kiosk shown in **FIG. 3A3**, except that the kiosk shown in **FIG. 3A6** is scaled down in size and reduced in weight to enable the device to be completely transportable within the store by the hand of the shopper, as taught hereinabove. As shown in **FIG. 3A6**, the communication connection between the transportable kiosk hereof and the infrastructure of the Internet **10A** can be realized using wireless digital communication technology (e.g. RF-based communication subsystems, using DFSK or spread-spectrum modulation techniques) well known in the art in order to provide (i) transportability within retail shopping environments for the convenience of shoppers, as well as (ii) Internet access to the IPI Web-site of the present invention (i.e. hosted on mirrored IPD Servers **11**).

[0223] As taught in the Objects and Summary of the Present Invention set forth hereinabove, the client computer system of the present invention **13** may also be realized in the form of a bar code driven multi-media kiosk mounted upon a conventional shopping cart, or other transportation vehicle, so as to be completely transportable within retail shopping environments for the convenience of consumers, as shown in **FIG. 3A7**. As taught hereinabove, the retail shopping environment may be a department store, super-

market, superstore, retail outlet or the like. Notably, the bar code driven shopping cart kiosk shown in **FIG. 3A7** is similar to the bar code driven multi-media kiosk shown in **FIG. 3A3**, except that the kiosk shown in **FIG. 3A7** is mounted upon a conventional shopping cart to be completely transportable within the store, as taught hereinabove. As shown in **FIG. 3A7**, each shopping cart supported kiosk hereof comprises a cart structure **90** having a basket portion, a push-type handle bar portion **92**, a set of wheels and a kiosk device **13** mounted to the handle bar portion. In this illustrative embodiment, the client computer subsystem embodied within the kiosk includes a 2-way RF communication link with its I/O port and a network hub associated with an IP network mounted within the retail shopping environment of concern. A wireless spread-spectrum communication subsystem including base station **98** and network adapter cards, such as the Symbol Spectrum24 wireless LAN (WLAN) by Symbol Technologies, Inc. can ,be used to realize the 2-Way RF communication link in a manner known in the communications network art. Such a wireless LAN provides (i) transportability with retail shopping environments for convenience of shoppers, as well as (ii) Internet access to the IPI Web-sites of the present invention (i.e. hosted on mirrored IPD Servers **11**). By strategically placing the access points within the retail environment, the Symbol Spectrum 24 wireless LAN makes it possible to create a wireless bridge between a wired (IP-based) LAN within the retail environment (operably connected to the Internet by an ISP) and any number of shopping cart supported kiosks, as well as fixed mounted kiosks, and transportable Internet access terminals lent to consumers for use within the retail shopping environment.

[0224] In alternative embodiments, any Client Computer **13** can be realized as a network computer (NC), a WebTV™ type Internet Terminal, a Newton MessagePad® PDA, or any other device providing Internet access to the IPI Web-site (i.e. mirrored IPD Servers) of the present invention. Notably, the same functionalities provided within the Web-based kiosk described above can be embodied with such alternative embodiments of client computer system.

[0225] For example, as shown in **FIG. 3A8**, the client computer subsystem **13** can be realized as a transportable hand-held computer, such as the Newton® Model 130 Messagepad **70** from Apple Computer, Inc. of Cupertino, Calif., provided with NetHopper™ brand Internet Access (http-client) Software which supports the TCP/IP networking protocol within the Newton MessagePad operating system, as well as the client-side of http, as taught in U.S Pat. No. 5,905,251 incorporated herein by reference. Notably, the NetHopper™ brand Internet Access (http-client) Software **71** provides the Newton Model 130 Messagepad with an integrated JAVA GUI-based web browser program for WWW access in a manner know in the Internet access art. As shown in **FIG. 3A8**, the Newton Messagepad has a display panel **72**, touch-screen type keypad **73**, and programmed laser scanning bar code symbol reader **74** (e.g., Metrologic ScanQuest® Laser Scanning Module Model No. IS4120), integrated within the hand-held device as described in U.S Pat. No. 5,905,251. The function of bar code symbol scanner **74** is to read UPC or UPC/EAN symbols on consumer products and to produce symbol character data representative of the numbers encoded within such standardized bar code structures. The Newton Messagepad Model 130, denoted by reference numeral **70**, is also equipped with a Motorola PCMCIA-based modem card **75** having a RF transceiver for establishing a wireless digital communication link with either a cellular base station or one or more satellite-base stations **76** connected to the Internet by way of an ISP or NSP **10A** in a manner well known in the global information networking art. As such, a first wireless digital communication link **77** is established between the Newton Messagepad 130 and cellular (or satellite) base stations **76**, and a second digital communications link **78** is established between the base station **76** and the ISP or NSP associated with the infrastructure of the Internet. Accordingly, this embodiment of the client computer subsystem of the present invention is completely mobile (i.e. transportable and provide the consumer access to the Internet and all of its information resources on the WWW and elsewhere, provided that the device maintains its wireless digital communication link with base station **76**, distributed through the globe, making access to the IPD servers hereof possible at home, in the office, within retail stores, as well as on the road wherever that may be.

[0226] As shown in **FIG. 3A8**, the Newton MessagePad, ScanQuest® Laser Scanning Module **74** and auxiliary battery supply (not shown) are completely housed within a rubberized shock-proof housing **79**, in order to provide a hand-supportable unitary device **70** of rugged construction. This hand-held Internet-enabled wireless information access terminal can be used virtually anywhere, provided wireless Internet access is enabled by digital IP communication network service providers (NSPs) in operation about the planet. Operation of Internet access terminal **70** is quite simple from the user's point of view. Upon reading a bar code symbol **80** on a consumer product **81**, the object detection field **82** of the device automatically detects the consumer product, and in response thereto, a laser beam **83** is automatically projected and swept across the UPC symbol **80** thereon. While it is generally preferred that the automatic laser scanning engine **74** be interfaced with I/O communications port of the Newton MessagePad device **70**, it is understood that, in some instances, it may be desired to connect a pen or wand-type scanning device to the serial port thereof to provide bar code symbol reading capabilities thereto. Optionally, bar code decoding software can be run on the Message Pad device, or as firmware contained within the scanning engine **74** in a manner known in the art.

[0227] Notably, it is understood that there will be many different types of wireless mobile Internet-enabled access terminals that may be used to realize the client computer subsystems of the present invention. For example, recently 3COM, Inc. introduced into its commercial product line the PalmPilot VII Wireless Hand-Held Internet Access Terminal, which is similar in many respects to the Newton Messagepad Model 130 equipped with the Motorola PCMCIA-based modem card **75**, and Nethopper™ Software, described above. Also, Symbol Technologies, Inc. of Holtsville, N.Y. has introduced the Symbol SPT 1500, SPT 1700, SPT 1740 and PPT 2700 hand-held wireless bar code scanning Internet access terminals which have virtually the same functionalities embodied within the wireless hand-held Internet access terminal shown in **FIG. 3A8**, and originally disclosed in U.S Pat. No. 5,905,251, supra.

[0228] The Web-enabled client subsystems **13** of the present invention described hereinabove may be used to access consumer product-related information, as well as to

carry out electronic-commerce related transactions, at home, at work, in the office, on the road, as well as in physical retail shopping environments.

[0229] For example, when visiting particular EC-oriented (i.e. electronic-commerce enabled) Web-sites, a consumer may scan UPC (and/or UPC/EAN) numbers on products within his or her home (e.g. in the pantry) using any one of the client computer subsystems hereof equipped with a bar code symbol Reader in order to remotely purchase such consumer products using credit or debit type financing, and direct shipment of purchased products to the consumer's home or elsewhere by a particular delivery service. Such EC-enabled WWW sites, commonly referred to as electronic-commerce (EC) stores or storefronts, as well as on-line electronic commerce catalogues, can be operated by manufacturers, wholesalers and/or retailers of consumer products, as indicated in FIGS. **2-1** and **2-2**. As shown therein, retailer operated, managed and/or owned EC stores (i.e. EC-enabled WWW sites) are hosted on retailer operated/owned EC information servers (MECIS) 12B, whereas manufacturer operated, managed and/or owned EC stores (i.e. EC-enabled WWW sites) are hosted on manufacturer operated/owned EC information servers (MECIS) 12B operably connected to the infrastructure of the Internet.

[0230] The consumer product information delivery system of the present invention shall enable an infinite array of applications with regard to electronic commerce and home shopping, now made possible by the present invention.

The Retail Store Based Consumer Product Information Transport Subsystem of the Present Invention

[0231] As shown in **FIG. 3A9**, each a retail shopping environment participating in the system of the present invention is provided with a TCP/IP local area network **80** comprising: a plurality of bar code driven kiosks **13** as shown in FIGS. **3A2** through **3A8**, having (i) a Web browser program **13A** (e.g. Netscape Communicator 4.5 for Windows 98 with a customized GUI) for enabling consumers to access and display information resources on the Internet (e.g. WWW), and (ii) an e-mail client program **133** (e.g. POP3 mail client software) **82** for supporting e-mail based CPI transport operations with remotely-situated e-mail-enabled client subsystems **13** connected to the Internet, in accordance with the principles of the present invention; a network information server **84** running (i) e-mail messaging software **84A** for maintaining e-mail accounts and service for each bar code driven kiosk on the retail store based LAN **80**, (ii) http server software **84B** for serving locally-stored Web document (e.g. advertisements, product prices, specials, notices, etc.) to consumers on the kiosks, as well as (iii) firewall software **84C** for maintaining network security; a TCP/IP router **86** connected between the network information server **84** the infrastructure of the Internet (i.e. ISP or NSP), for connecting the retail store based LAN and its connected kiosks to the Internet. Notably, the TCP/IP router **86** is assigned a static IP address that determines the IP address for the retail store based LAN **80**. Also, each kiosk can be assigned a static IP address on the retail store based LAN, or a dynamically allocated IP address using the well know Dynamic Host Computer Protocol (DHCP), enabling both Web and e-mail services on each kiosk.

[0232] In general, the network information server **84** can be realized by, for example, the Origin 200 Server or O_2 Desktop Workstation from Silicon Graphics, Inc., a high-end information server from Sun Microsystems, Inc., or any other computing machine that can perform the function of a server in a web-based, client-server type computer system architecture of the illustrative embodiment. Exemplary electronic messaging (i.e. e-mail) software solutions for the network information server **84** include, but are not limited to: N-Plex® Global Internet Messaging Server solution from Isocor, Inc. of Santa Monica, Calif.; the Message Direct Server from Messaging Direct, Inc.; Mail Spinner, from Nascent; Netscape Messaging Server, from Netscape Communications, Inc.; or Sendmail Pro, from Sendmail Inc. computer system architecture of the illustrative embodiment. Exemplary http server software solutions for the network information server **84** include, but are not limited to: the Java Web Server (JWS) 1.0 or later from JavaSoft, division of Sun Microsystems, Inc.; the JigSaw Web Server from the World Wide Web Consortium; the Fastrak™ Web (http) server from Netscape Communications, Inc.; the Internet Information Server (IIS) from the MicroSoft Corporation; the Apache HTTP Server from The Apache Software Foundation; or any other http server capable of transporting HTML-encoded documents.

[0233] Alternatively, each network information server **84** can be realized using the Whistle INTERJET II network information server solution from IBM as a turn-key solution for the network information server **84** on the retail store LAN **80**. Details on the Whistle INTERJET II server can be found at http://www.whistle.com, incorporated herein by reference. This implementation can support up to hundred (100) bar code driven kiosks on a retail store LAN of the present invention, and in some applications, it may be desirable to modify the e-mail software provided thereon in order to achieve the business objectives of any particular application. It is understood, however, that in many application, in which advertisements, prices and specials, notices and the like are to be displayed on the kiosks during idle moments (i.e. when consumers are not scanning bar coded products for consumer product related information access and display), there will be a need to use a more robust electronic messaging and http server solutions on the retailer's network information server **84**.

[0234] As shown in **FIG. 3A10A**, a preferred way of implementing the retailer based information network of in **FIG. 3A9** would be to install a wireless LAN within each "brick & mortar" retail shopping environment, thereby enabling TCP/IP network connectivity between each Web/e-mail enabled kiosk **13** and the infrastructure of the Internet. As shown in **FIG. 3A10A**, each Web/e-mail enabled kiosk **13** is seamlessly connected to the TCP/IP network of the retailer LAN **80** using high data rate wireless LAN, such as the Spectrum24™ High Rate Wireless LAN (WLAN) from Symbol Technologies, Inc., of Holtsville, N.Y. Notably, the Spectrum24 High Rate WLAN is designed to the proposed IEEE 802.11 open airwaves standards including the Ethernet backbone, TCP/IP protocols, SNMP network management, PCMCIA adapter card form factors and interfaces, and NDIS and compliant drivers. According to this wireless solution, each Web/e-mail enabled kiosk **13** is equipped with either a Spectrum24™ wireless LAN PC card, radio card, or ISA card **90A** to enable a wireless TCP/IP connection to the retail based LAN within the store environment. A Spectrum24 network controller **90B**, with integrated RF antenna elements, is installed within the retail

shopping environment, preferably in a central location which facilitates excellent RF signal transmission/reception between the network controller **90**B and the LAN PC card **90**A in each of the Web/e-mail enabled kiosks **13** mounted within the retail shopping environment on, for example, a product self **2000** shown in FIGS. **3A10**B through **3A10**C. As shown in **FIG. 3A10**A, the network controller **90**B and network information server **84** are connected to a TCP/IP hub **92** configured within the LAN according to a suitable connectivity, well known in the art. In turn, the TCP/IP hub **92** is connected to the infrastructure of the Internet (i.e. ISP) using a modem and a communication link in a manner known in the art. In turn, the TCP/IP hub **92** is connected to the TCP/TP routers **86**, which is connected to the Internet infrastructure Notably, using the Spectrum24 network controller **90**, the system administrator can enable administration and configuration of the Web/e-mail enabled kiosks, RF traffic control, node management, and diagnostics. Notably, the 2 Mbps or 11 Mbps version of the Spectrum24 High Rate WLAN can support Voice-over-IP applications in instances where telephonic hand-sets are provided at the kiosk, as shown in **FIG. 3A3** through **3A7**.

[0235] As shown in **FIG. 3A9**, the central e-mail server **88** is assigned a static IP address and connected to the infrastructure of the Internet using a modem in a manner known in the art. The retailer RDBMS **81** and a manufacturer RDBMS **83** are connected to the backend of the central e-mail server **88** by way of a high-speed TCP/IP network **91**. Preferably, RDBMS **89**A and **89**B support protocols such as XML/ICE to enable electronic data interchange with client machines operated by retailers and manufacturers alike. The information server supporting the central e-mail server **88** may also support a http server and a suitable database interface to enable retailer and manufacturers alike to access the RDBMS **89**A and RDBMS **89**B over the Internet using XML, EDI, ftp or other information interchange protocol.

[0236] FIGS. **3A10**B through **3A10**C illustrate an exemplary product shelving system in a retail shopping environment, wherein a plurality of a web/e-mail enabled bar code driven CPI kiosks **12** are mounted onto shelf structures **99** by way of a kiosk mounting bracket **13**C that can be removably attached to a shelf structure **99** (using a special tool) as well as disattached therefrom for reinstallation within the parts of the retail store as circumstances require. As shown, each web/e-mail enabled kiosk **13** is equipped with a wireless LAN PC card **90**A in order to establish a wireless connection with retailer LAN **90** via the wireless LAN controller **90**B. As shown in FIGS. **3A10**B, the kiosk **13** depicted therein is a modified version of the kiosk shown in FIGS. **3A3** and **3A4**, wherein the bar code reading device **36'** is pivotally mounted on the kiosk housing and orientated for optimal product label scanning.

[0237] In order to better understand the functions of the centralized e-mail server **88** and its back-end RDBM **89**A and **89**B illustrated in FIGS. **3A9** and **3A10**, it will be helpful to provide a brief overview of the CPI transport services enabled by the e-mail based information transport subsystem described above.

[0238] During operation of the IPI Finding and Serving Subsystem hereof within a retail shopping environment, the consumer having accessed and displayed a consumer product related Web document on a bar code driven consumer product information kiosk as shown, for example, in FIGS. **3A2** through **3A8**, may also desire to retain a copy thereof for future reference and use. In such instances, it would desirable to provide the bar code driven information kiosk **13** with a thermal or like printer so that consumers can printout accessed product related information within the retail shopping environment and take the same home for future review and evaluation. However, from the retailer's point of view, providing each such CPI kiosk with a printer may be too costly to maintain in typical retail environments. Thus, there is a great need for a n improved method of and system for making consumer copies of consumer product information that has been accessed and displayed on bar code driven consumer product information kiosks within retail shopping environments.

[0239] In accordance with the principles of the present invention, the problem of providing consumers with copies of accessed consumer product information within retail shopping environments is addressed by enabling the consumer at the retail-based kiosk to: (1) display an e-mail envelope (ready for stuffing, addressing and sending) the display frame **20**C of the Web browser program thereof, by manually selecting control button **21**G provided along the control frame **20**B, shown in **FIG. 3A14**A; (2) capturing, saving, and attaching any accessed/displayed consumer product document to the displayed e-mail envelope by manual (or voiced-directed) selection of the "capture, save and attach" button **110** within the displayed e-mail envelope of **FIG. 3A14**, or capturing and recording the URL of the CPI-related document being displayed by manual (or voice-directed) selection of the "capture and record" button **112** within the displayed e-mail envelope of **FIG. 3A16**; (3) addressing the e-mail envelope with the consumer/shopper's home, office or like e-mail address by either reading an e-mail address encoded within a bar code (or magnetic-stripe) structure or manually entering the same within the addressee field; and (4) sending the stuffed e-mail envelope by manual selection of the "send" button **114** within the displayed e-mail envelope. The enabling infrastructure for this e-mail enabled consumer product information transport subsystem will be described hereinbelow.

[0240] Notably, the e-mail envelope displayed on each information kiosk hereof, as shown in FIGS. **3A14** and **3A16**, need not indicate that a copy of a particular e-mail message is being sent to the centralized e-mail server **88** upon selecting the "send" button, although circumstances may dictate that notice be given to customers using this e-mail CPI-related transport service within retail shopping environments. In the case where the "cc" field visually indicated to the consumer, it may be desirable to enable the consumer to delete preset recipients in the addressee fields thereof so that, upon transmission, no copies of transmitted e-mail envelope will be sent to third parties (e.g. retailers and/or manufacturers), thereby providing the customer with a greater sense of confidentiality and privacy with respect to its product inquires when using this e-mail CPI-related transport service.

[0241] Having provided an overview of the functions of the e-mail CPI-related transport service of the present invention, it is appropriate at this juncture to briefly describe the primary functions to be performed by central e-mail server **88** and RDBMs **89**A and **89**B shown in **FIG. 3A9**.

[0242] In the context of the e-mail CPI-related transport service of the present invention, the primary function of the central e-mail server **88** shown in FIGS. **3A9** and **3A10** is to receive a "carbon copy" (Cc) of each e-mail envelope sent from a retailer kiosk within the system of the present invention, to the e-mail address of the shopper (or friend thereof) accessible at home, in the office, at school, or on the road. In alternative embodiments of the present invention, the central e-mail server **88** can be realized as a mirrored array of e-mail servers connected to different points of the Internet about the planet, whose main purpose is to collect copies of e-mail CPI transmissions sent to the e-mail addresses of consumers/shoppers from Web/e-mail enabled kiosks **13** within retail shopping environments. Such information is important to retailers and manufacturers as it reflects the consumer product related interests of consumers shopping at particular "brick & mortar" retail stores, located at particular geographic regions on the Earth. Potentially, each such region will have a different market significance to particular retailers and/or manufacturers.

[0243] Once such CPI-related information has been collected by the central e-mail server(s) **88** the retailer RDBMS **81** periodically downloads a copy of the e-mail CPI transmission records maintained within the central e-mail server subsystem **88**. Thereafter, such records are processed and reorganized in a form that is readily useful to retailers who offer particular UPN-labeled products for sale. An exemplary database structure for the retailer RDBMS **89A** is set forth in **FIG. 3A11**. Preferably, the retailer RDBMS **89A** will contain information on which Web documents (specified at a particular URL) were accessed at a particular retailer kiosk (having a geographic specification) and transported to a particular e-mail address of a consumer considering the purchase of the UPN-labeled product linked to the URL Notably, the retailer RDBMS **89A** of the illustrative embodiment can be realized as an Octane® Workstation or O_2 Desktop Workstation from SGI of Mountain View, Calif., a high-end SUN information server from Sun Microsystems, Inc., or any other high-end computing platform running RDBMS solution software such as, for example, Oracle8i (Release 2) RDBMS software from Oracle Corp. of Redwood Shores, Calif., and data mining and analysis software such as, for example, Mineset 3.0 Visual data mining software from SGI. Using such data mining and analysis tools, various types of reports can be generated for individual retailers indicating trends in consumer shopping behavior, as well as the address and identity of prospective customers. Preferably, only retailers registered with the system will be provided access to information maintained within the retailer RDBMS **89A** that relate to Web documents accessed and reviewed by the retailer's kiosks, prior to or after making a consumer product search and/or purchase within a particular retail shopping environment.

[0244] Similarly, the function of the manufacturer RDBMS **83** is to periodically download a copy of the e-mail CPI transmission records maintained within the central e-mail server subsystem **88**, and to thereafter process and reorganize the same so as to be put into a form that will be readily useful to manufacturers who make or have made particular UPN-labeled products for sale and publish particular CPI-related Web documents on the WWW and link the same to particular UPN-labeled products. An exemplary database structure for the manufacturer RDBMS is set forth in **FIG. 3A12**. Preferably, the manufacturer RDBMS **89B** will contain information on which Web documents (specified at a particular URL) were accessed at a particular retailer kiosk (having a geographic specification) and transported to a particular e-mail address of a consumer considering the purchase of the UPN-labeled product linked to the URL. Notably, the manufacturer RDBMS **89B** of the illustrative embodiment can be realized as an Octane® Workstation or O_2 Desktop Workstation from SGI of Mountain View, Calif., a high-end SUN information server from Sun Microsystems, Inc., or any other high-end computing platform running RDBMS solution software such as, for example, Oracle8i (Release 2) RDBMS software from Oracle Corp. of Redwood Shores, Calif., and data mining and analysis software such as, for example, Mineset 3.0 Visual data mining software from SGI. Using such data mining and analysis tools, various types of reports can be generated for individual manufacturers indicating trends in consumer shopping behavior, as well as e-mail leads on prospective customers. Preferably, only manufacturers registered with the system will be provided access to information maintained within the manufacturer RDBMS **89B** that relate to Web documents accessed and reviewed by shoppers or the retailer's kiosks, prior to or after making a particular product search and/or purchase within a particular retail shopping environment.

[0245] Having described the structure and function of the e-mail information transport subsystem of the present invention, it is appropriate now to describe is method and modes of operation hereinbelow.

The Database Structure of the IP) Server

[0246] In the illustrative embodiment of the present invention, each data-synchronized IPD Server **11** of the preferred embodiment maintains at least two different relational-type databases, namely: a IPI Registrant Database for storing information about manufacturers whose products are registered with the system; and a Non-IPI Registrant Database for storing information about manufacturers whose products are not registered with the system. A schematic representation of the IPI Registrant Database is shown in **FIG. 4A1**, whereas a schematic representation of the Non-IPI Registrant Database is shown in **FIG. 4B**.

[0247] As shown in **FIG. 4A1**, the relational-type IPI Registrant Database maintained by each IPD Server comprises a plurality of labeled information fields for each product "registered" therewith, namely: an IPN Information Field for storing information (e.g. numeric or alphanumeric string) representative of the Universal Product Number (e.g. twelve-digit UPC Version A number, eight-digit UPC Version E number, thirteen-digit UPC/EAN number, or twelve-digit UPC Version A number plus five-digit Add-On Code Segment number frequently used in the publishing industry) assigned to the consumer product; a Company Name Information Field for storing information (e.g. numeric or alphanumeric string) representative of the name of the company making, selling or distributing the corresponding product; a URL Information Field(s) for storing information (e.g. numeric or alphanumeric string) representative of the Universal Resource Locator (URL) or Universal Resource Locators (URLs) at which information resource(s) of the multi-media type can be found on the Internet relating to the corresponding consumer product; a Trademark Information Field for storing information (e.g. text and/or alphanumeric

strings) representative of each trademark (or Domain Name) used in connection with the promotion, sale, distribution and/or use of the corresponding product, and preferably registered with the United States Patent and Trademark Office (USPTO) or other governmental or quasi-governmental agency (e.g. INTERNIC or Network Solutions, Inc.); a Product Description Information Field for storing information (e.g. text strings) descriptive of the corresponding product; an E-mail Address Information Field for storing information (e.g. numeric or alphanumeric string) representative of the e-mail address of the corresponding company (e.g. manufacturer) on the Internet; a CPIR-Enabling Applet Information Field for storing information representative of consumer product information request (CPIR) enabling Applets accessible by retailers, wholesalers, advertisers, Web publishers and the like by downloading operations to be described in detail hereinafter, and eventually inserted within the HTML code of Web documents on various types of Internet information servers used to host WWW sites of all sorts, so that, when executed, these CPIR-enabling Applets automatically access from the master UPN/URL Database Management Subsystem **9** hereof, a categorized menu of URLs specifying the location of information resources on the Internet pertaining to a particular UPN-labeled product and symbolically linked thereto by its manufacturer or authorized agent; image file storage field for storing color images of consumer products registered with the system; and a Status Information Field for storing information (e.g. numeric or alphanumeric string) representative of whether the company (e.g. manufacturer) associated registered product has paid their monthly, quarterly or annual registration fees associated with registration within the IPD Servers of the information finding and serving subsystem hereof. Notably, each information item contained with the information field shown along the same horizontal line of **FIG. 4A1** is related or linked.

[0248] In general, the URL stored in the URL Information Field specifies the address of an information resource on the Internet (e.g. Web), and thus may point to any one of the following types of information resources: a HTML document or file on the World Wide Web (expressed in the HyperText Markup Language); a single record in a database; the front-end of an Internet program such as Gopher; or the results of a query made using another program. In accordance with convention, the syntactic structure of each URL generally comprises: a Protocol Specifier, such as "http", "ftp", "gopher", "news", or "mail to", and specifies the type of resource that the URL is pointing (i.e. connecting) to; a Host Indicator, represented by double slashes "//" if the URL is requesting information from a Web Server; Server Name comprising an Internet Domain Name (e.g. "www."), the address of the Web Server (e.g. "ibm."), and a designator (e.g. "com", "edu", "int", "mil", "org", etc.) identifying who owns the server or where it is located; a Path Name, such as "Products/Computers/", indicating a path to the destination information file on the identified Server; and a Resource Name (including file extension, e.g. ".html"), such as "aptiva.html", identifying the actual named information file that contains actual information resource specified by the URL.

[0249] As used herein as well as in the Claims to Invention, the term "registered" and the variants thereof shall be understood to mean listed or having an entry within a database. Such listing or entry can be achieved in a variety of ways including, but not limited to: (i) by specific request of the associated company or business; or (ii) by the system administrator without a request and/or authorization of the corresponding company or business linked to the product.

[0250] Notably, each information item contained within the information field shown along the same horizontal line of **FIG. 4A1** is symbolically related or linked. Different products of the same registrant or related registrant may also be linked together so that a user looking for information about a particular product is automatically provided with URLs which are assigned to related products of the registrant which may satisfy the goals or objectives of a particular advertising and/or marketing campaign or product promotion program of the registrant company. As it may be desired to relate particular products at particular points in time, the relationships therebetween can be dynamically changed within the IPI Registrant Database by a straightforward database updating operation carried out by a system administrator (or manager) who, in theory, can be located virtually anywhere throughout the world. Expectedly, such database updating operations would be carried out using appropriate system access and security procedures well known in the art.

[0251] Inasmuch as the UPC data structure is presently employed as a universal product identifier (i.e. a primary data structure) in a majority of industries throughout the world, its twelve-digit numeric string (for UPC Version A) or eight-digit numeric string (for UPC Version E) will be a preferred UPN (in many applications) for purposes of carrying out the principles of the present invention. This twelve (12) digit human-readable number, printed on the bottom of each UPC label (and encoded within the bars and spaces of the UPC label itself), comprises: (i) a six digit manufacturer number assigned to the manufacturer by the Uniform Code Council, Inc. (UCC) of Dayton, Ohio, and consisting of a one digit "number system" number and a five digit manufacturer code; (ii) a five digit "product" number assigned to the product by the manufacturer; and (iii) a one digit modulo check digit (mathematically calculated) and added to each UPC number to check that the code has been read correctly by the bar code symbol reader.

[0252] In order to provide the requester greater control over what information is actually displayed on its client subsystem, the URL Information Field of the IPI Database shown in **FIG. 4A1** contains a number of information subfields. As shown in **FIG. 4A2**, these information subfields comprise: a Product Advertisement Information Field for storing information representative of URLs pointing to information on the Internet relating to advertising and/or promotion of the product; a Product Specification (i.e. Description) Information Field for storing information representative of URLs pointing to information on the Internet relating to specifications on the product; a Product Update Information Field for storing information representative of URLs pointing to information on the Internet relating to product updates, recalls, notices, etc; a Product Distributor (e.g. Wholesaler and/or Resaler) Information Field for storing information representative of URLs pointing to information on the Internet relating to distribution, sale and/or ordering of the product; a Product Warranty/Servicing Information Field for storing information representative of URLs pointing to information on the Internet relating to warranty, extended warranty offerings, servicing and maintenance of the product; a Product Incentive Information Field (e.g. rebates, discounts and/or coupons) for storing information

representative of URLs pointing to information on the Internet relating to rebates, discounts and sales on the product; a Product Review Information Field for storing information representative of URLs pointing to information on the Internet relating to reviews, analysis, testing, inspection and/or comparison of the product; and Miscellaneous Information Field(s) for storing information representative of URLs pointing to information on the Internet relating to miscellaneous aspects of the product (e.g., direct product sales on the WWW, product installation/set-up and operating manuals, company reports (10 Ks, annual reports, etc.), and the like. Each URL symbolically linked to a UPC-labeled product registered in the Registered IPI Database is categorized within one or more of these URL categories.

[0253] The list of URLs recordable in the IPI Registrant Database for each registered UPC-labeled product is virtually unlimited. Below are just a few examples of how the IPI Finding and Serving Subsystem hereof can be used as a virtual sales agent that provides value-added services to consumers, retailers and the like.

[0254] For each CD sound recording, the URL list may contain a URL that points to a promotional QuickTime® video recording or MP3-formatted sound recording published on the WWW for reviewing and evaluation by the consumer. The promotional song can be by a commissioned or endorsing artist, as is typically done in conventional advertising programs. The same can be done for video recordings on tape and digital video discs (DVDs). The URL may also provide the consumer with a down-loadable trial version of the product for a limited time period.

[0255] For each computer software product, the URL list may contain a URL that points to a multi-media clip on the WWW that provides a demonstration of the solutions that the software product provides, as well as the functions and development tools that it enables. It may also provide the consumer with a down-loadable version of the software product for a time-limited trial period.

[0256] For electronic consumer products, the URL list may contain a URL that points to a multi-media clip on the WWW that provides an audio-visual demonstration of the product in various user environments. Also, the URL list can contain a URL that points to a Web-based Specification Sheet that can be printed out in a retail environment, at home, work or on the road.

[0257] For groceries and like articles, the URL list may contain a URL that points to a multi-media clip on the WWW that provides a QuickTime® video recording or the like of the product, illustrating various cooking recipes and uses for the product. Also, the URL list can contain a URL that points to a Web-based Discount Coupon that can printed out in the store, at home or work.

[0258] For toys, the URL list may contain a URL that points to a multi-media clip on the WWW that provides an audio-visual demonstration of the toy along with promotional endorsements by the various characters used in its advertising campaign.

[0259] For clothing, garments, or accessories (e.g. wearing apparel), the URL list may contain a URL that points to a multi-media clip on the WWW that provides a QuickTime® video recording or the like of the clothing, garments, and/or accessories being modeled by stunning fashion models. Ideally, such video recordings, linked to particular articles of wearing apparel by their UPC number, can be used to extend and augment the advertising campaign being carried out in other forms of media (e.g. television, radio, print, billboards, etc.).

[0260] Preferably, the manufacturer, its marketing personnel and advertising agents will actively participate in the creation of the product related information resources, as well as the placement of their URLs into the above-defined (or like) URL categories maintained within the Database of the IPI Finding and Serving Subsystem hereof. Also, using the Manufacturer/Product Registration Subsystem hereof, manufacturers and/or their agents can easily link their UPNs (e.g. UPC and/or EANs) with such URLs and manage the same in a dynamic manner to ensure that product related information on the Internet is accurately linked to the UPNs of the manufacturer's products. Through such active participation, the business objectives of any particular manufacturer or retailer can be promoted by way of the IPI Finding and Serving Subsystem of the present invention. In this way, the information requesting consumer is provided with only the kinds of product-related information which he or she seeks.

[0261] As shown in **FIG. 4B**, the Non-IPI Registrant Database maintained by each IPD Server comprises a plurality of labeled information fields for each product that is not currently registered with the IPD Server, namely: an IPSN (i.e. IPN) information Field for storing information (e.g. numeric or alphanumeric string) representative of the Universal Product Number (e.g. a UPC number from a UPC numbering system, or an EAN numbering system) assigned to the non-registered product; a Company Name Information Field for storing information (e.g. numeric or alphanumeric string) representative of the name of the company making, selling or distributing the corresponding non-registered product; a Trademark Information Field for storing information (e.g. text and/or alphanumeric strings) representative of each trademark used in connection the promotion, sale, distribution and/or use of the corresponding product, and preferably registered with the USPTO or other governmental agency; a Product Description Information Field for storing information (e.g. text strings) descriptive of the corresponding product; and an E-mail Address Information Field for storing information (e.g. numeric or alphanumeric string) representative of the e-mail address of the corresponding company (e.g. manufacturer) on the Internet; a Status Information Field for storing information (e.g. numeric or alphanumeric string) representative of whether the company associated non-registered product has been solicited by the IPD Server, and on what dates registration solicitation has occurred. Notably, each information item contained with the information field shown along the same horizontal line of **FIG. 4A1** is related or linked. The information required to construct the Non-IPI Registrant Database shown in **FIG. 4B** can be readily obtained from a number of commercially or publicly available information sources (e.g., the Universal Code Council, Inc., Dayton, Ohio; QRS, Inc. Of Richmond. Calif.; General Electric Information Services (GEIS) of Delaware, Md.; etc.).

Constructing the IPI Registrant Database Within the IPI Finding and Serving Subsystem Hereof

[0262] The utility of the product finding functionalities of the IPI Finding and Serving Subsystem hereof depends in

large part on the number of consumer-products registered with the IPI and Non-IPI Registrant Databases supported within the UPN/URL Database Management Subsystem **9** hereof, as illustrated in FIGS. **2B1, 2B2, 2B3** and **2B4**. In principle, numerous techniques may be employed separately or in combination with each other in order to construct the IPI and Non-IPI Registrant Databases of the present invention. Six such techniques will be detailed below.

[0263] According to a first database construction technique, the administrator of the IPI Registrant Database would transmit Product Registration Requests (PRRs) in the form of electronic documents to each and every the manufacturer having been issued, for example, a six digit UPC Manufacturer Identification Number (MIN) by the UCC, Inc. Such electronic documents can be transmitted using conventional MIME protocols such as, for example, STMP. The Product Registration Request document would seek to ascertain from the manufacturers the various information items (including the menu of URLs) identified in the IPI Registrant Database of **FIG. 4A1**. In response to the Product Registration Request, each solicited manufacturer would send back to the administrator of the IPI Registrant Database (for each of its consumer products) its UPC number and a menu of categorized URLs indicating the location of the information resources identified in the Product Registration Request document. This information can then be used to readily construct the IPI Registrant Database of the illustrative embodiment.

[0264] According to a second database construction technique, a global advertising campaign would launched (over various media) in order to solicit the various information elements identified in the IPI Registrant Database of **FIG. 4A1** and thus register the products of the manufacturers selling UPC-labeled products. Preferably, such information would be collected by way of an electronic data transfer subsystem(s) set-up to cooperate with the system of the present invention in order to facilitate database construction operations.

[0265] According to a third database construction technique, the IPI system itself would continuously solicit consumer product registrations over time in order to collect information from companies responding favorably to the solicitations. While such solicitation efforts can involve the issuance of product registration requests using various types of media, it is preferred that the information collection operations are carried out using electronic data transfer techniques (e.g. ftp, EDI or XML/ICE) described hereinabove.

[0266] According to a fourth database construction technique, a number of commercial on-line Internet search engines, such as AltaVista™, Yahoo™, WebCrawler™, Lycos™, Excite™, as well as powerful off-line parallel-processing search engines, would be enlisted to analyze (i.e. mine) information on the WWW in order to collect and link the information elements specified in the IPI Registrant Database of **FIG. 4A1**.

[0267] Once an "initial" IPI Registrant Database has been constructed using an y one or more of the four database construction techniques described hereinabove, manufacturers registered therewith can be periodically contacted using Web-based electronic document (i.e. message) transfer techniques in order to request updating and confirmation of the UPN/URL listings contained within the database of the IPI subsystem of the present invention.

[0268] According to a fifth database and preferred construction technique of the present invention, the Registrant IPI Database of the system would be initially "seeded" with several items of information obtained and related without the assistance of manufacturers of UPC-labeled products. Such information items include: (1) the six digit UPC Manufacturer Identification Numbers encoded in the UPC symbols (and numbers) applied to the products of such UCC-registered manufacturers; and (2) the URLs of the Web home pages of such manufacturers.

[0269] The first step of this database construction method involves obtaining the six digit Manufacturer Identification Numbers (MINs) uniquely issued to manufacturers by the Uniform Code Council, Inc. of Dayton, Ohio. Such MINs can be obtained from various commercial sources including GE Information Services, QRS, Inc. formerly Quick Response Services, Inc.), as well as the UCC. At present, about 180,000 Manufacturers Identification Numbers have been issued to manufacturers by the UCC. A string of six zeros (i.e. 000000) may be added to each one of these 180,000 or so six digit Manufacturer Identification, Numbers in order to produce 180,000 or so 12 digit numbers (i.e. hereinafter referred to as "Manufacturer's Reference Numbers) for the 180,000 or so manufacturers listed in the IPI Registrant Database under construction. As each such Manufacturer Reference Number has the same length as a UPC number of its manufacturer, this number can be conveniently thought of as the "Manufacturer Reference UPC Number" which can be stored in the UPN Information Field of the Database along with the corresponding manufacturers name being stored in the Company Name Information Field.

[0270] The second step of the method involves finding the URL of the Web home page of each of the 180,000 or so manufacturers who, to date, have been assigned a Manufacturers Identification Code and are listed in the Database. Such URL information can be found using a number of available techniques: (i) using a commercially available search engine to search the WWW in order to find the URL of the home page of each manufacturer's Web-site, if it has one, using the name and address thereof obtained during the first step above; or (ii) using a commercially available (INTERNIC-enabled) Domain Name search service that uses the names and addresses of the manufacturers (obtained during the first step above) in order to determine whether a particular manufacturer has a registered domain name on the Internet, and if so, is the domain name being actively used in a URL that points to the home page of the manufacturer's Web-site. Once obtained, such URLs are then added to the IPI Database, along with the e-mail and/or other address of the manufacturer symbolically linked thereto (if available).

[0271] Having constructed the "seeded" Database, it can then be used to connect the client subsystem of users to the home page of Web-sites of manufacturers of particular products. Initially, when an Internet user provides the UPC number of a particular product as input to the Input Box of the HTML form displayed in the information display frame of the client subsystem (e.g. when operated in its Manufacturer Website Search Mode), then the IPD Server need only compare the first six digits of the entered UPC number against the first six-digits of the Manufacturer Reference

UPC Numbers (i.e. Manufacture Identification Numbers) listed in the "seeded" Database. The corresponding (home-page) URL of the matching manufacturer is returned to the client subsystem C_a for display. In instances of an initially seeded Database, in which only the "Manufacturer Reference UPC Numbers" are listed therein, the requesting client subsystems are provided with the URLs of the home pages of the symbolically linked manufacturers. Then as manufacturers begin to register their consumer products with the system (e.g. in response to mass e-mailings, advertisements and/or marketing and promotional efforts, etc.), the IPD Database will return a menu of "hot-linked" URLs, for each registered product, pointing to various types of product-related information resources on the Internet (described above) that can be easily accessed by simply clicking thereon in a conventional manner. Over time, Manufacturer Reference UPC Numbers and the URLs of the "home pages" of such manufacturers will become replaced by the UPC numbers of registered products and the menu of URLs on the WWW symbolically linked thereto by the manufacturers, thereby allowing consumers and users of the system to precisely pinpoint consumer product-related information on the WWW which has been specified by the manufacturer, its marketing department and/or advertising agency. With manufacturer's and advertiser's participation and feedback, the initially seeded RDBMS described hereinabove will quickly grow into a robust RDBMS richly filled with the various information items described in FIGS. **4A1** and **4A2**, including the symbolically linked UPCs and URLs that point to very specific consumer product related information resources (i.e. files) stored within IPI Servers of the system located across the global expanse of the Internet.

[0272] According to a sixth database construction technique of the present invention, the Registrant IPI Database of the system would be constructed by allowing each manufacturer to construct a limited or restricted version of the master UPN/URL Database (i.e. Registrant IPI Database) of the system, wherein only UPC-encoded products of the registered manufacturer and Web-based information items related thereto are entered into the database. As will be described in greater detail herebelow in connection with the third method of Product Registration in the subsystem hereof, the creation of such limited-version UPN/URL databases can be carried out by providing each registered manufacturer with a computer program that allows its administrators to construct and manage a limited UPN/URL database in a "turn-key" manner. Also, from its Website, the manufacturer can serve the limited UPN/URL database over the Internet to consumers. As part of the registration process, each registered manufacturer transmits its limited UPN/URL database to Web-server which then integrates all such databases in order to update the master UPN/URL database (IPI Registrant Database) of the system.

[0273] Methods of Constructing UPN-Encoded Server-Side and Client Side Applets and Distributing the HTML Tags Associated Therewith to Remote Client Subsystems for Embedding Within HTML-Encoded Documents to be Published Over the Internet in Connection With UPN-Labeled Consumer Products

[0274] In general, for each system architecture shown in FIGS. **2B1** through **2B4**, there will be a different Applet-driven method used to access consumer product related information (e.g. UPN/URL links, trademark/URL links, etc.) from the RDBMS server **9** and display the search results within a Java-based GUI at the point of presence of the consumer using a Java-enabled client machine **13**. In order to practice these different methods, in various e-commerce related applications which will be described in greater detail hereinafter, it will be helpful to construct either server-side or client-side UPN-encoded CPIR-enabling Applets, distribute the HTML tags associated therewith to remote client subsystems, and thereafter embed these CPIR-enabling Applet tags within HTML-encoded documents for publishing over the Internet. Such CPIR-enabling Applet construction, distribution and embedding methods will now be described in detail hereinbelow.

The First Applet-Driven Method of Accessing and Displaying Categorized UPN/URL Link Menus From the UPN/URL Database Management Subsystem the Present Invention

[0275] After providing a brief overview on the system architecture of **FIG. 2B1** and the nature of the server-side CPIR-enabling Applets deployed therewithin, The steps associated with Applet-driven CPI-acquisition method of the first illustrative embodiment will be described in detail with reference to FIGS. **4E1**, **4E2**, **4F1** and **4F2**.

[0276] In general, the method of FIGS. **4E1** and **4E2** involves using a server-side CPIR-enabling Applet to automatically conduct a UPN-directed search on the UPN/URL Database Management Subsystem **9** hereof (i.e. RDBMS server **9**) in response to a single mouse-clicking operation by the consumer on the HTML tag associated with the server-side Applet. In the illustrative embodiment, the CPIR-enabling servlet of the present invention is a program written in the Java™ programming language and has an HTML tag (indicated by <SERVLET>) which is designed to be included in an HTML page, much in the same way an image can be included therewithin (according to the HTML 3.2 Specification).

[0277] CPIR-enabling servlets of the present invention are designed to work within a request/response processing model, as shown in **FIG. 2B1**. In this request/response model, a client subsystem **13** sends a request message to the Java Web Server **11'** and the Server **11'** responds by sending back a reply message. In the illustrative embodiment, requests come in the form of http, although is understood that the use of other protocols such as ftp, EDI or a custom protocol, may be possible in particular embodiments. The request and the corresponding response reflect the state of the client and the server at the time of the request.

[0278] When using a Java-enabled browser to view a Web page containing a server-side CPIR-enabling Applet tag <SERVLET>, the servlet's compiled class code is automatically accessed from the Java Web Server **11'** and executed on the server-side of the network connection illustrated in **FIG. 2B1**. Thus, as shown in **FIG. 2B1**, the Java Web Server **11'** must provide a Java Virtual Machine (JVM) for running/executing Java servlets on the server-side of the network in much the same way that a client browser must provide a JVM for running Java Applets on the client side thereof. Additionally, the Java Web browser **11'** must also support the Java Servlet API, developed by JavaSoft, and define how and when the servlet communicates with the Java Web Server. Essentially, the Servlet API is a well-defined set of

function calls (i.e. set of Java classes) to get information to and from the Java Web Server. The servlet needs to be able to access server-defined variables, issue redirects, send error messages and the like. Sun's Java Web Server supports the Servlet API and, of course, includes their JVM.

[0279] As shown in FIG. 2B1, the Java Web Server 11' includes a number of software components including the Java Servlet API which comprises several Java interfaces and fully defines the link between the hosting server (e.g. Java Web Server) and the servlets located at the middle tier. The Servlet API is defined as an extension to the standard JDK. This means that there is a n explicit definition of servlet interfaces, but it is not part of the Java Development Kit (JDK) 1.1 or the Java 2 platform. Instead, the servlet classes are delivered with the Java Servlet Development Kit (JSDK) version 2.0 from Sun (http://java.sun.com/products/servlet/). This JSDK version is intended for use with both JDK 1.1 and the Java 2 platform. There are a few significant differences between JSDK 2.0 and JSDK 1.0.

[0280] JDK extensions are packaged under javax—the root of the Java extension library tree. The Java Servlet API contains the following packages: Package javax.servlet; and Package javax.servlet.http dedicated to supporting HTTP protocol and HTML generation. The Servlet API provides a tight link between a server and servlets, allowing servlets to add new protocol support to a server. Essentially, any protocol (e.g. SMTP, POP, FTP, etc.) that follows a request/response computing model can be implemented by a servlet.

[0281] General servlet support is provided by the package "javax.servlet" which comprises the following components:

[0282] (1) Servlet: An interface that defines communication between a web server and a servlet. This interface defines the init( ), service( ), and destroys( ) methods (and a few others).

[0283] (2) ServletConfig: An interface that describes the configuration parameters for a servlet. This is passed to the servlet when the web server calls its init( ) method Note that the servlet should save the reference to the ServletConfig object, and define a getServletConfig( ) method to return it when asked. This interface defines how to get the initialization parameters for the and the context under which the servlet is running.

[0284] (3) ServletContext: An interface that describes how a servlet can get information about the server in which it is running. It can be retrieved via the getServletContext( ) method of the ServletConfig object.

[0285] (4) ServletRequest: An interface that describes how to get information about a client request.

[0286] (5) ServletResponse: An interface that describes how to pass information back to the client.

[0287] (6) GenericServlet: A base servlet implementation. It takes care of saving the ServletConfig object reference, and provides several methods that delegate their functionality to the ServletConfig object. It also provides a dummy implementation for init( ) and destroy( ).

[0288] (7) ServletInputStream: A subclass of InputStream used for reading the data part of a client's request. It adds a readLine( ) method for convenience.

[0289] (8) ServletOutputStream: An OutputStream to which responses for the client are written.

[0290] (9) ServletException: Should be thrown when a servlet problem is encountered.

[0291] (10) UnavailableException: Should be thrown when the servlet is unavailable for some reason.

[0292] Support for HTTP Servlets is provided by the package "javax.servlet.http" which comprises the following components:

[0293] (1) HttpServletRequest: A subclass of ServletRequest that defines several methods that parse HTTP request headers.

[0294] (2) HttpServletResponse: A subclass of ServletResponse that provides access and interpretation of HTTP status codes and header information.

[0295] (3) HttpServlet: A subclass of GenericServlet that provides automatic separation of HTTP request by method type. For example, an HTTP GET request will be processed by the service( ) method and passed to a doGet( ) method.

[0296] (4) HttpUtils: A class that provides assistance for parsing HTTP GET and POST requests.

[0297] The central abstraction in the Java Servlet API is the Servlet interface. All servlets implement this interface, either directly or more commonly by extending a class that implements it (e.g. such as the HttpServlet class). The Servlet interface declares but does not implement methods that manage the servlet and its communication with clients. The servlet writer will provide some or all of these methods when developing a CPIR-enabling servlet.

[0298] Having provided an overview on server-side CPIR-enabling Java Applets (i.e. Servlets) of the present invention and the support framework required thereby in a distributed-computing object oriented programming environment shown in **FIG. 2B1**, it is appropriate to now describe the method of creating, loading, distributing, embedding and executing server-side CPIR-enabling Java Applets in accordance with the principles of the present invention schematically illustrated in FIGS. **4E1, 4E2, 4F1** and **4F2**.

[0299] As indicated at Block **A1** in **FIG. 4E1**, the first step of the method involves using the Java Servlet API to write or otherwise author the source code for a server-side CPIR-enabling Java Applet., for each UPN-specified consumer product registered in the UPN/URL Database Management Subsystem **9**. In general, the source code for each server-side CPIR-enabling Java Applet (i.e. servlet) will embody one or more of following items of information, namely: (i) the UPN of the particular product on which the CPI search is to be carried out and the search results thereof displayed; (ii) Java classes required for performing a UPN-directed search on the RDBMS Server **9** using one or more Java methods running natively on the Java Web Server **11'**, and producing a particular Java GUI for displaying the results obtained from the UPN-directed search; and (iii) license-related information specifying the terms and conditions of the CPIR-enabling Servlet license and the conditions under which the CPIR-enabling servlet shall operate.

[0300] Notably, such license-related information may specify: (1) one or more specific host domains (e.g.

www.homedepot.com or www.walmart.com) from which a Web document containing the corresponding servlet tag may launch the CPIR-enabling servlet under a licensing program; (2) one or more general Internet domains (e.g. .com, org, .gov, .int, .mil, .uk, etc.) from which a Web document containing the corresponding servlet tag may launch the CPIR-enabling servlet under a licensing program; (3) the time duration of the licensing period associated with the CPIR-enabling servlet; and (4) any other restrictions set by the associated manufacturer and/or retailer, and/or administrator of the consumer product information system hereof, that must be observed for a registered CPIR-enabled servlet to operate within a Web-document served from the registered Internet domain.

[0301] Notably, the Java source code for each CPIR-enabling Servlet will vary depending upon implementation. However, regardless of the particular implementation, it can be expected that each JDBC-supporting CPIR-enabling servlet when, for example, designed to search an (Oracle JDBC-supported) UPN/URL Database Server **11A** for the UPC/URL list currently symbolically linked to a specified UPN and display the search results on the requesting client machine **13**, will typically include Java source code specifying:

[0302] (1) the importable JDBC classes required by the CPIR-enabling Servlet;

[0303] (2) the importable Java classes to be used in the CPIR-enabling Servlet;

[0304] (3) the JDBC driver to be loaded for the Oracle-based UPN/URL Database;

[0305] (4) the connection strings to the UPN/URL Database;

[0306] (5) the CPI query to be executed on the UPN/URL Database, dependent on the UPN of the associated consumer product and possibly other search criteria and Servlet licensing conditions;

[0307] (6) the servlet tag, its graphical icon or alias to trigger execution the Servlet and its associated CPI query;

[0308] (7) the CPI Search Result GUI to be displayed on the requesting client machine and its relative location to the associated Servlet tag; and

[0309] (8) the operations that will be carried out upon execution of the CPI query including

[0310] Boolean search logic to be carried out upon initiation of the UPN-directed CPI search;

[0311] if a new connection is required between Java Web Server and the UPN/URL Database;

[0312] Loading the JDBC driver;

[0313] Connecting to the UPN/URL Database;

[0314] Creating a SQL statement based on the specified Boolean search logic and UPN;

[0315] Executing the SQL query statement; and

[0316] Dumping the search results to the CPI Search Result GUI.

[0317] In the embodiment depicted in **FIG. 2B1**, a UPN-encoded Servlet is used to replace a Common Gateway Interface (CGI) script and provide a way of searching the UPN/URL Database on RDBMS Server **11'**, with the advantage of increased speed and stability. In this instance, CPIR-enabling Servlets are accessed by the user as an HTML tag <SERVLET> embedded in an HTML document served to the client subsystem **13**. For example, when the consumer selects the servlet tag (graphically encoded by an icon or image) in an HTML-encoded document, the linked servlet residing on the server-side of the network, is automatically executed, causing a UPN-directed search to be carried on the RDBMS server **9**.

[0318] After writing/authoring the source code for the Applet, the CPIR-enabling Java servlet is ascribed a unique name such as, for example, "UPNXXXXXXYYYYYZ" for a 12 digit Uniform Product Code.

[0319] Reference can be made to the following technical publication for additional details on writing Java servlets, designing Java GUIs and the like: "Java Servlets and Serialization With RMI" (1999) by Scott McPherson, published by Sun Microsystems, Inc., of Palo Alto, Calif.; "The Java™ Tutorial Second Edition: Object Oriented Programming For The Internet (1999), by Mary Campione and Kathy Walrath, published by Sun Microsystems. Inc., of Palo Alto, Calif.; and "The JFC Swing Tutorial: A Java Guide To Constructing JAVA GUIs" (1999), by Mary Campione and Kathy Walrath. published by Sun Microsystems, Inc., of Palo Alto, Calif.; "JAVA" (1997) be Ed Tittel and Bill Brogden, published by IDG Books Worldwide, Inc.; "Wilde's WWW: Technical Foundations of the World Wide Web" (1999) by Erik Wilde, published by Springer-Verlag, Berlin, Heidelberg each said publication being incorporated herein by reference.

[0320] As indicated at Block **A2** in **FIG. 4E1**, the source code for the authored servlet is compiled into Java bytecode, and java bytecode for the servlet classfiles are placed in the server_root/servlets directory on the Java Web Server **11'**. When compiling, the javax.servlet.* package should be placed in the classpath. The easiest way to do this is to include server_root/lib/classes.zip in the classpath, but it is understood that one must look to the compiler's documentation for specific details in this regard.

[0321] As indicated at Block **B1** in **FIG. 4E1**, the second step of the method involves using the Java Server Administration Applet to configure the Java Web Server so as to extend the functionalities thereof and embody (or install) the CPIR-enabling Java servlet within the Java Web Server **11'**. This process of extending the functionalities of the Java Web Server **11'** involves specify the default parameters and arguments thereof. This configuration step is carried out when using the Servlet Loading facility of the Administration Applet in order. The loading process is achieved by clicking on the Servlets Button, and then selecting Add from the list of choices on the left GUI of the Servlet Loading Facility. Thereafter, to add a new servlet, the following procedure is performed: Enter in the Servlet Name field, the unique name for the CPIR-enabling Java servlet one is loading onto the Java Web Server (e.g. "UPNXXXXXX-YYYYYZ"); Enter in the Servlet Class field, a valid class name for the Java class of the CPIR-enabling servlet, i.e., the full package name, e.g. "sun.server.http.FileServlet"; and thereafter Click on the Add button.

[0322] As indicated at Block B2 in **FIG. 4E1**, the fourth step of the method involves invoking the CPIR-enabling Java servlet by creating a URL having the path section "/servlet/" prepended to the assigned Servlet Name, so that the URL can be thereafter embodied within the servlet HTML tag <SERVLET>, prior to its insertion within the HTML code of a Web document. To invoke a servlet, the webmaster/administrator calls the servlet by creating a URL with "/servlet/" prepended to the servlet name. One can confirm that the servlet is correctly invoked by entering this URL into ones Web browser and analyzing the output of the created servlet.

[0323] As indicated at Block B3 in **FIG. 4E1**, the fifth step of the method involves (1) embodying the unique URL, created for each consumer product, within a CPIR-enabling servlet HTML tag <SERVILE>, (2) containing each such servlet HTML tag within an executable file, and (3) storing each such servlet tag containing file in the Central CPIR-Enabling Applet Library on the RDBMS Server **9**.

[0324] As indicated at Block C in **FIG. 4E2**, the sixth step of the method involves distributing the CPIR-enabling servlet HTML tags (within the Central CPIR-enabling Applet Library) to retailers, wholesalers, advertisers, and others who desire to deliver UPN-directed CPI search results to their customers, clients and the like. This distribution process can be carried out by downloading servlet tag containing files to a remote computer system connected to the Internet using, for example, ftp or other electronic data or document interchange protocols (e.g. XML/ICE) to carry out the servlet tag transport process. These downloaded CPIR-enabling servlet tags can then be stored in a local CPIR-enabling Applet/Servlet Library maintained on a client computer **13**, as shown in **FIG. 4F2**, until it is time to embed the same into a particular HTML-encoded document.

[0325] As indicated at Block D in **FIG. 4E2**, the seventh step of the method involves enabling retailers, wholesalers, advertisers, and others to (1) open the downloaded servlet tag containing files, (2) extract the CPIR-enabling servlet tags contained therewithin, and (3) embed (i.e. insert) one or more distributed CPIR-enabled servlets HTML tags into acceptable HTML-encoded documents associated with EC-enabled WWW sites, EC-enabled storefronts and catalogs, Internet product advertisements, on-line auction-based WWW sites, or other types of Web-documents.

[0326] In general, this step of the method involves first creating or otherwise procuring a suitable HTML-encoded document which may, understandably include other types of code (e.g. XML) therein, other than HTML code. While such HTML documents can be created using any HTML-editing program, such as BBD-Edit, it is expected that in most applications the underlying HTML-encoded document will be generated using tools such as, for example: GO-LIVE® WWW-Site Development and Management solution software from Adobe Systems, Inc. to create the HTML pages associated with a particular WWW site; Catalog-Maker™ and CatalogManager electronic commerce solution software programs from RealEDI, Inc; Intershop **4** Enfinity™ Electronic Commerce Solution software from Intershop Communications, Inc; and/or any other commercially available HTML-authoring tools which enable quick and easy creation of HTML-encoded documents, and easy insertion of any downloaded CPIR-enabling servlet tag <SERVLET> using, for example, simple commands or drag-and-drop procedures.

[0327] As indicated at Block E in **FIG. 4E2**, the eighth step of the method involves serving "servlet-tag" encoded HTML documents from Internet information servers to Java-enabled client computer subsystems **13** operated by consumers at home, in the office, in EC-enabled or "brick and mortar" retail stores, or on the road, as the case may be. As shown in **FIG. 4F**, such Internet information servers can include, for example, IPI servers **12**, retailer-related EC-enabled information servers **12A**, manufacturer-related EC-enabled information servers **12B**, and/or any other Internet (http or ftp) information servers operating on the Internet from which HTML-encoded document are served for any informational, educational, and/or entertainment purpose.

[0328] As indicated at Block F in **FIG. 4E2**, the ninth step of the method hereof involves using a Java-enabled client computer subsystems **13** to display served HTML-encoded documents having one or more of CPIR-enabling servlet tags embedded therewithin. This step is carried out by the consumer pointing his or her Java-enabled browser program (e.g. Netscape Navigator, Microsoft Explorer, or Sun Microsystems' HotJava program) to an HTML-encoded document within which a CPIR-enabling Java servlet tag is embedded, at a particular point of presence on the WWW. As shown in FIGS. **4M1** through **4R2**, CPIR-enabling Applets can be graphically-encoded in an variety of different ways to provide the consumer with a visual indication that, clicking on the graphical object however manifested, will automatically result in a consumer product information search on a particular product identified by the UPN encoded within the associated servlet. In view of the fact the CPIR-enabling servlet tags are distributed over the Internet and inserted within HTML documents by others than the servlet author, at some future date, graphical encoding of CPIR-enabling servlets will typically occur at the time of writing the servlet.

[0329] With the above point in mind, it will be helpful to adopt a standardized icon for graphically indicating the presence of a CPIR-enabling servlet tag within an HTML document. In the illustrative embodiments shown in FIGS. **4P1**, **4P2**, **4R1**, **4R2**, **4S1**, and **4S2**, small predefined images of servicemarks such as "GO: UPC Request Cyber-Service™ URL Search" are served to inform the consumer that the Java object, if selected from the displayed Web page, will automatically cause a product-specific URL search to be performed with respect to the particular consumer product and the results thereof displayed at the "point of presence" of the consumer who may be residing at a particular point in an EC-enabled store (e.g. at the check-out display screen or POS), at on-line auction site, at a Web-based product advertisement, or anywhere else on the WWW. Notably, an important advantage provided by this information search technique of the present invention is that it does not disturb the consumer at his or her point of presence (or sale), where ever that may be. Instead, the CPI search and display method hereof enables the delivery of accurate product-specific manufacturer-defined information at precise points in Cyberspace by performing a single mouse-clicking operation. This enables consumers to make informed decisions thereat based on the information displayed in the corre-

sponding Java GUI generated upon launching a CPIR-enabling servlet at the consumer's point of presence on the WWW.

[0330] It is understood, however, that other techniques may be used to create a visual indication to the consumer that a CPIR-enabling Applet is located at a particular point on the WWW and that if this Applet is executed (e.g. by a single mouse-clicking operation), then a UPN-directed consumer product information search will be automatically executed and the results therefrom will be displayed within a Java GUI at the consumer's point of presence.

[0331] One alternative technique would be to embed within the CPIR-enabling Applet, a thumb-nail or large size photo-image of the consumer product being offered for sale, lease, auction, or other purpose on the WWW. Notably, this product image any reside on the RDBMS server **9**, or on the http server from which the HTML-encoded document is served. Using this technique, the consumer need only click on the image to initiate a UPN-directed consumer product information search against the UPN/URL Database Management Subsystem **9** hereof.

[0332] Notably, the person or persons responsible for delivering product advertisements to particular locations on one or more WWW sites can use the OPEN ADSTREAM™ (OAS) 5.0 Internet Advertisement Management Solution software from Real Media, Inc., of New York, N.Y., and any other suitable software solution, running on the Internet (http) information server (**12**, **12'**, **12A** or **12B**), and managed using an Web-enabled client subsystem **13**, as shown in **FIG. 4F1**. Using the OAS 5.0 advertisement management solution, and the CPI search and display method of present invention described above, a webmaster or advertising manager assigned to a particular Internet information server (**12**, **12'**, **12A** or **12B**) can: (1) access the Web-based product advertisement for a particular product (i.e. HTML code, image files, and any other rich media content associated therewith); (2) access previously downloaded CPIR-enabling Java servlet(s) for the consumer product, stored in a locally-maintained "Library (i.e. Catalog) of CPIR-Enabling Applets" on a client machine or server on the network, or directly access CPIR-enabling Java servlets from the centrally-maintained "Library (i.e. Catalog) of CPIR-Enabling Applets" illustrated in **FIG. 4F1**, via the Java Web Server **11'**; and (3) use Real Media's OAS 5.0 solution software deliver both the Web-based product advertisement (i.e. its HTML code and other media-rich content) and the corresponding CPIR-enabling Applet tag (and possibly any image files associated therewith) to a designated section on a particular Web page of a specific WWW-site.

[0333] It is understood that there are different ways of inserting/embedding both of these objects within a particular section of an HTML-encoded document using the highly-advanced HTML-editing functionalities of the OAS 5.0 software system. For example, the Web-based product advertisement can be inserted within a first spatially-defined portion of the target HTML document (occupying the largest portion of the purchased Internet advertising space), while the CPIR-enabling server-side Applet (i.e. servlet) is inserted within a second spatially-defined portion of the target HTML document occupying the balance of the purchased Internet advertising space. Alternatively, both the CPIR-enabling Applet and the Web-based product advertisement can be inserted within substantially same spatially-defined portion of the target HTML document so as to achieve spatial overlap therebetween. This way when the consumer clicks on the advertisement image, or some preselected portion thereof, the underlying CPIR-enabling servlet will be automatically executed and the corresponding Java GUI generated for displaying the results of the UPN-directed database search.

[0334] In situations where the Internet product advertisement (e.g. banner advertisement) embodies a servlet HTML tag which, when executed, produces a new Java GUI (i.e. new browser interface), then a CPIR-enabling servlet can be embedded within the HTML-encoded document displayed in the new Java GUI. This servlet tag embedding technique will be useful in many applications where the display space allocated for the Web advertisement in the target HTML-encoded document is limited, and there is a need to generate a new Java GUI for presenting the content of the advertisement.

[0335] Other ways of embedding the Web-based advertisement and the related CPIR-enabling servlet tags will become apparent hereinafter to those skilled in the art having had the benefit of reading the present disclosure.

[0336] As indicated at Block G in **FIG. 4E2**, the tenth step in the method involves the consumer recognizing that a CPIR-enabling servlet tag is embedded within a Web-document displayed on a Java-enabled client computer subsystem, and thereafter launching/executing the associated servlet to initiate a UPN-directed consumer product information search within the RDBMS server **9**.

[0337] Notably, the above illustrative embodiment has been described with particular focus given to CPIR-enabling servlets encoded with the UPN of a particular consumer product. It is understood, however, that the CPIR-enabling servlets of the present invention can be encoded with the trademark(s) used in connection with a particular consumer product, thus providing Trademark-encoded CPIR-enabling servlets, in contrast with UPN-encoded CPIR-enabling servlets. In such alternative embodiments, the encoded trademark would be used to direct a search through the RDBMS server **9**, and display the results thereof in a new (independent) Java GUI generated at the point of servlet tag embodiment. Alternatively, a product descriptor associated with a particular product can be encoded within the corresponding CPIR-enabling servlet, used to direct a search through the RDBMS server **9**, and display the results thereof in an independent Java GUI generated at the point of servlet tag embodiment.

The Second Applet-Driven Method of Accessing and Displaying Categorized UPN/URL Link Menus From the UPN/URL Database Management Subsystem the Present Invention

[0338] After providing a brief overview on the system architecture of **FIG. 2B2** and the nature of the client-side CPIR-enabling Applets deployed therewithin, the steps associated with Applet-driven CPI-acquisition method of the second illustrative embodiment will be described in detail with reference to FIGS. **4G1**, **4G2**, **4H1** and **4H2**.

[0339] In general, the method of FIGS. **4G1** and **4G2** involves using a client-side CPIR-enabling Applet to auto-

matically conduct a UPN-directed search on the UPN/URL Database Management Subsystem hereof (i.e. RDBMS server **9**) in response to a single mouse-clicking operation by the consumer on the HTML tag associated with the CPIR-enabling Applet. In the illustrative embodiment, the CPIR-enabling Applet of the present invention is a program written in the Java™ programming language and has an HTML tag (indicated by <APPLET>) which is designed to be included in an HTML page, much in the same way an image can be included therewithin (according to the HTML 3.2 Specification).

[0340] CPIR-enabling Applets of the present invention are designed to work within a request/response processing model, as shown in **FIG. 2B2**. In this request/response model, a client subsystem **13** sends a request message to the Java Web Server **11"** and the Server **11"** responds by sending back a reply message. In the illustrative embodiment, requests come in the form of http, although is understood that the use of other protocol as such as ftp, EDI or a custom protocol, may be possible in particular embodiments. The request and the corresponding response reflect the state of the client and the server at the time of the request.

[0341] When using a Java-enabled browser to view a Web page containing a client-side CPIR-enabling Applet tag <APPLET>, the Applet's compiled class code is automatically accessed from the Java Web Server **11"** and executed on the client-side of the network connection illustrated in **FIG. 2B2**. Thus, as shown in **FIG. 2B2**, the Java-enabled client machine **13** in this network architecture must run a Java-enabled browser program that provides a Java Virtual Machine (JVM) for running/executing Java Applets on the client-side of the network in much the same way that Java Web Server **11"** must provide a JVM for running Java servlets on the server side thereof, as shown in **FIG. 2B1**.

[0342] As shown in **FIG. 2B2**, the Java-enabled Web browser **13** includes a number of software components including Java interfaces for fully defining the link between the Java Web browser and the Applets located at the middle tier. In order to write, compile and load Applets onto the Java Web Server **11"**, the system administrator or webmaster can use the Java API provided for within the Java 2 (development) platform from JavaSoft, a division of Sun Microsystems, Inc. This platform also supports The API specification of the Java 2 Platform, Standard Edition, version 1.2.2, comprises the following Packages:

[0343] (1) java.applet: Provides the classes necessary to create an applet and the classes an applet uses to communicate with its applet context.

[0344] (2) java.awt: Contains all of the classes for creating user interfaces and for painting graphics and images.

[0345] (3) java.awt.color: Provides classes for color spaces.

[0346] (4) java.awt.datatransfer: Provides interfaces and classes for transferring data between and within applications.

[0347] (5) java.awt.dnd: Drag and Drop is a direct manipulation gesture found in many Graphical User Interface systems that provides a mechanism to information between two entities logically associated with presentation elements in the GUI.

[0348] (6) java.awt.event: Provides interfaces and classes for dealing with different types of events fired by AWT components.

[0349] (7) java.awt.font: Provides classes and interface relating to fonts.

[0350] (8) java.awt.geom: Provides the Java 2D classes for defining and performing operations on objects related to two-dimensional geometry.

[0351] (9) java.awt.im: Provides classes and an interface for the input method framework.

[0352] (10) java.awt.image: Provides classes for creating and modifying images.

[0353] (11) java.awt.image.renderable: Provides classes and interfaces for producing rendering-independent images.

[0354] (12) java.awt.print: Provides classes and interfaces for a general printing API.

[0355] (13) java.beans: Contains classes related to Java Beans development.

[0356] (14) java.beans.beancontext: Provides classes and interfaces relating to bean context.

[0357] (15) java.io: Provides for system input and output through data streams, serialization and the file system.

[0358] (16) java.lang: Provides classes that are fundamental to the design of the Java programming language.

[0359] 17) java.lang.ref: Provides reference-object classes, which support a limited degree of interaction with the garbage collector.

[0360] 18) java.lang.reflect: Provides classes and interfaces for obtaining reflective information about classes and objects.

[0361] 19) java.math: Provides classes for performing arbitrary-precision integer arithmetic (BigInteger) and arbitrary-precision decimal arithmetic (BigDecimal).

[0362] (20) java.net: Provides the classes for implementing networking applications.

[0363] (21) java.rmi: Provides the RMI package.

[0364] (22) java.rmi.activation: Provides support for RMI Object Activation.

[0365] (23) java.rmi.dgc: Provides classes and interface for RMI distributed garbage-collection (DGC).

[0366] (24) java.rmi.registry: Provides a class and two interfaces for the RMI registry.

[0367] (25) java.rmi.server: Provides classes and interfaces for supporting the server side of RMI.

[0368] (26) java.security: Provides the classes and interfaces for the security framework.

[0369] (27) java.security.acl: The classes and interfaces in this package have been superseded by classes in the java.security package.

[0370] (28) java.security.cert: Provides classes and interfaces for parsing and managing certificates.

[0371] (29) java.security.interfaces: Provides interfaces for generating RSA (Rivest, Shamir and Adleman AsymmetricCipher algorithm) keys as defined in the RSA Laboratory Technical Note PKCS#1, and DSA (Digital Signature Algorithm) keys as defined in NIST's FIPS-186.

[0372] (30) java.security.spec: Provides classes and interfaces for key specifications and algorithm parameter specifications.

[0373] (31) java.sql: Provides the JDBC package.

[0374] (32) java.text: Provides classes and interfaces for handling text, dates, numbers and messages in a manner independent of natural languages.

[0375] (33) java.util: Contains the collections framework, legacy collection classes, event model, date and time facilities, internationalization, and miscellaneous utility classes (a string tokenizer, a random-number generator, and a bit array).

[0376] (34) java.util.jar: Provides classes for reading and writing the JAR (Java ARchive) file format, which is based on the standard ZIP file format with an optional manifest file.

[0377] (35) java.util.zip: Provides classes for reading and writing the standard ZIP and GZIP file formats.

[0378] (36) javax.accessibility: Defines a contract between user-interface components and an assistive technology that provides access to those components.

[0379] (37) Javax.swing: Provides a set of "lightweight" (all-Java language) components that, to the maximum degree possible, work the same on all platforms.

[0380] (38) javax.swing.border: Provides classes and interface for drawing specialized borders around a Swing component.

[0381] (39) javax.swing.colorchooser: Contains classes and interfaces used by the JcolorChooser component.

[0382] (40) javax.swing.event: Provides for events fired by Swing components.

[0383] (41) javax.swing.filechooser: Contains classes and interfaces used by the JfileChooser component.

[0384] (42) javax.swing.plaf: Provides one interface and many abstract classes that Swing uses to provide its pluggable look-and-feel capabilities.

[0385] (43) javax.swing.plaf.basic: Provides user interface objects built according to the Basic look-and-feel.

[0386] (44) javax.swing.plaf.metal: Provides user interface objects built according to the "metal" look-and-feel.

[0387] (45) javax.swing.plaf.multi: The multiplexing look and feel allows users to combine auxiliary look and feels with the default look and feel.

[0388] (46) javax.swing.table: Provides classes and interfaces for dealing with java.awt.swing.JTable.

[0389] (47) javax.swing.text: Provides classes and interfaces that deal with editable and noneditable text components.

[0390] (48) javax.swing.text.html: Provides the class HTMLEditorKit and supporting classes for creating HTML text editors.

[0391] (49) javax.swing.text.html.parser

[0392] (50) javax.swing.text.rtf: Provides a class (RTFEditorKit) for creating Rich-Text-Format text editors.

[0393] (51) javax.swing.tree: Provides classes and interfaces for dealing with java.awt.swing.JTree.

[0394] (52) javax.swing.undo: Provides support for undo/redo capabilities in an application such as a text editor.

[0395] (53) org.omg.CORBA: Provides the mapping of the OMG CORBA APIs to the Java™ programming language, including the class ORB, which is implemented so that a programmer can use it as a fully-functional Object Request Broker (ORB).

[0396] (54) org.omg.CORBA.DynAnyPackage: Provides the exceptions used with the DynAny interface (InvalidValue, Invalid, InvalidSeq, and TypeMismatch).

[0397] (55) org.omg.CORBA.ORBPackage: Provides the exception InvalidName, which is thrown by the InconsistentTypeCode, which is thrown by the Dynamic Any creation methods in the ORB class.

[0398] (56) org.omg.CORBA.portable: Provides a portability layer, that is, a set of ORB APIs that makes it possible for code generated by one vendor to run on another vendor's ORB.

[0399] (57) org.omg.CORBA.TypeCodePackage: Provides the user-defined exceptions BadKind and Bounds, which are thrown by methods in the class TypeCode.

[0400] (58) org.omg.CosNaming: Provides the naming service for Java IDL.

[0401] (59) org.omg.CosNaming.NamingContext-Package: Provides the exceptions used in the package org.omg.CosNaming (AlreadyBound, CannotProceed, InvalidName, NotEmpty, and NotFound) and also the Helper and Holder classes for those exceptions.

[0402] Having provided an overview on client-side CPIR-enabling Java Applets of the present invention and the support framework required thereby in a distributed-computing object oriented programming environment shown in **FIG. 2B2**, it is appropriate to now describe, the method of creating, loading, distributing, embedding and executing client-side CPIR-enabling Java Applets in accordance with the principles of the present invention schematically illustrated in FIGS. **4G1**, **4G2**, **4H1** and **4H2**.

[0403] As indicated at Block A1 in **FIG. 4G1**, the first step of the method involves using the Java Applet API to write or otherwise author the source code for a client-side CPIR-enabling Java Applet for each UPN-specified consumer product registered in the RDBMS server **9**. In general, the source code for each client-side CPIR-enabling Java Applet (i.e. Applet) will embody one or more of following items of information: (i) the UPN of the particular product on which the CPI-based search is to be carried out and the search results thereof displayed; (ii) Java classes required for performing a UPN-directed search on the RDBMS server **9** using a CGI script executing on the Java Web Server **11**", and producing a particular Java GUI for displaying the results obtained from the UPN-directed search; and (iii) license-related information specifying the terms and conditions of the CPIR-enabling Applet license and conditions under which the CPIR-enabling Applet shall operate.

[0404] Notably, such license-related information may specify: (1) one or more specific host domains (e.g. www.homedepot corn or www.walmart.com) from which a Web document containing the corresponding Applet tag may launch the CPIR-enabling Applet under a licensing program; (2) one or more general Internet domains (e.g. .com, .org., .gov, .int, .mil, .uk, etc.) from which a Web document containing the corresponding Applet tag may launch the CPIR-enabling Applet under a licensing program; (3) the time duration of the licensing period associated with the CPIR-enabling Applet; and (4) any other restrictions set by the associated manufacturer and/or retailer, and/or administrator of the consumer product information system of the present invention, that must be satisfied for a registered CPIR-enabled Applet to operate within a Web-document.

[0405] Notably, the Java source code for each CPIR-enabling Applet will vary depending upon implementation. However, regardless of the particular implementation, it can be expected that each CPIR-enabling Applet designed, for example, to search an (Oracle-based JDBC) RDBMS Server **9** for a current categorized UPC/URL list/menu symbolically linked to a specified UPN and thereafter display the results in an independent Java GUI, will typically include Java source code specifying:

[0406] (1) the importable JDBC classes required by the CPIR-enabling Applet;

[0407] (2) the importable Java classes to be used in the CPIR-enabling Applet;

[0408] (3) the JDBC driver to be loaded for the Oracle-based RDBMS server **9**;

[0409] (4) the connection strings to the RDBMS server **9**;

[0410] (5) the CPI query to be executed on the UPN/URL Database, dependent on the UPN of the associated consumer product and possibly other search criteria and Applet licensing conditions;

[0411] (6) the Applet tag, its graphical icon or alias to trigger execution the Applet and its associated CPI query;

[0412] (7) the CPI Search Result GUI to be displayed on requesting client and its relative location to the associated applet tag; and

[0413] (8) the operations that will be carried out upon execution of the CPI query including

[0414] Boolean search logic to be carried out upon initiation of the UPN-directed CPI search;

[0415] if a new connection is required between Java Web Server **11**" and the RDBMS server **9**;

[0416] Loading the JDBC driver;

[0417] Connecting to the RDBMS server **9**;

[0418] Creating a SQL statement based on the specified Boolean search logic and UPN;

[0419] Executing the SQL query statement; and

[0420] Dumping the search results to the CPI Search Result GUI.

[0421] When using earlier versions of the HTML Specification (i.e. HTML 3.2 by the World Wide Web Consortium), the source code for each CPIR-enabling Java Applet would adhere to the following general syntax:

[0422] [CODEBASE=codebaseURL]

[0423] CODE-appletFile

[0424] [ALT=alternateText]

[0425] [NAME=appletInstanceName]·

[0426] WIDTH=pixels HEIGHT=pixels

[0427] [ALIGN alignment]

[0428] [VSPACE=pixels][HSPACE=pixels]

[0429] >

[0430] [<PARAM NAME=Attribute1 VALUE= value >]

[0431] [<PARAM NAME=Attribute2 VALUE= value >]

[0432] . . .

[0433] [alternateHTML]

[0434] </APPLET>

[0435] wherein the set of brackets [] indicates optional parameters within the HTML 3.2 Specification.

[0436] The optional <PARAM>tag lets you to specify applet-specific attributes that your applet can retrieve as Strings. These Strings can be used in an HTML document to customize the Applet's behavior and supply their respective values in command-line variables.

[0437] CODEBASE, CODE, WIDTH, and HEIGHT are attributes specified by the first part of the <APPLET> tag. The Java-enabled Web browser **13** uses these attributes to locate the CPIR-enabling Applet code on the Java Web Server **11**" and to indicate the amount of space to be reserved in the target HTML document during display. CODEBASE directs the Java-enabled Web browser to look for code in the classes directory of the Java Web Server **11**". The mandatory parameter CODE provides the name of the Java code file that is to be loaded from the Java Web Server **11**". Any other class files used by the Applet are loaded from the Codebase Directory as well.

[0438] WIDTH and HEIGHT are attributes that tell the Java-enabled Web browser to reserve space in the document display before the Java code is loaded ("x" pixels wide by "x" pixels high), similar to the way browsers deal with images, reserving space before the complete image is available. To place an invisible Applet on a page, specify height and width of zero.

[0439] Current Java-enabled Web browsers ignore the Java language resize( ) method for applets. Although the applet viewer provided in the JDK responds to the resize( ) method, one will have to specify width and height correctly in the CPIR-enabling APPLET tag for general Web browsers.

[0440] VSPACE and HSPACE are attributes for specifying the amount of space (in pixels) that surrounds the CPIR-enabling Applet above and below (vspace) and on each side (hspace) of the Applet area.

[0441] ALIGN is another attribute used to designate where the Java-enabled Web browser is to place the Applet area in relation to any other design elements. ALIGN can have possible values like those for the IMG tag: left, right, top, texttop, middle, absmiddle, baseline, bottom, and absbottom.

[0442] Notably, the optional parameters ALIGN, VSPACE and HSPACE may be set either at the time of CPIR-enabling Applet creation (i.e. when writing the source code therefor), or at the latter time such as, for example, after compiling and loading the Java class code on the Java Web Server, after downloading CPIR-enabling Applet tag files, or after extracting CPIR-enabling Applets, but prior to embedding the CPIR-enabling Applet tag within a HTML-encoded document. Thus, these optional parameters enable the Web page, EC-commerce site and auction site designer to modify (after Java class code compiling and loading operations, but before Applet tag insertion/embedding operations) certain parameters and attributes within each CPIR-enabling Applet tag that determine the precise location where the Java-enabled browser on the client machine **13** will display the CPIR-enabling Applet and its associated image graphics on the browser display screen.

[0443] The ALT attribute designates a string to be displayed if the browser understands the <APPLET> tag but does not have Java capabilities. Netscape 2.0 allows turning the Java interpreter on or off via a check box under Options on the Security Preferences menu. The ALT attribute is a way to remind users to turn Java back on after turning it off to avoid wasting time on slow applets (as can often occur on Web pages).

[0444] CPIR-enabling Applets that coexist on the same Web page communicate by means of the NAME attribute. The AppletContext method getApplet(String name) returns a reference to the named Applet if it can be found in the Web page and then allows communication between Applets in the conventional sense.

[0445] The PARAM tag is used in transferring text information to the Applet code by way of the getParameter(String param_name) applet method. Since the parameter-matching code is not case sensitive and quote marks are parsed out, the following tags are equivalent:

[0446] <PARAM name=EMAIL

[0447] <param NAME=EMAIL

[0448] <param name="e-mail"

[0449] Between the last <PARAM> tag and the </APPLET> tag, text and HTML code can be designated for display if the Web browser of a particular client system cannot recognize the <APPLET> tag. Notably, the content of the Applet tag acts as alternate information for client subsystems that do not support this element or are currently configured not to support Applets. Otherwise, the content of the Applet must be ignored.

[0450] The final element of the Applet tag must be </applet> in order to tell the Web browser that the Applet has ended.

[0451] Notably, in the HTML 4.0 Specification, published by the World Wide Web Consortium at http://www.w3.org/TR/WD-htm140-970917/, the Applet element has been deprecated in favor of the OBJECT element, which offers an all purpose solution to generic object inclusion. The HTML 4.0 Specification now allows the OBJECT element to include images (via the <IMG> tag) and Applets (via the <APPLET> tag) in the same manner. Thus, when constructing Web documents and CPIR-enabling Applets in accordance with the HTML 4.0 Specification, the source code for each CPIR-enabling Applet will adhere to an entirely different syntax, the details of which are set forth which in the HTML 4.0 Specification, supra, are incorporated herein by reference.

[0452] For additional details pertaining to Java Applet construction, embedding techniques and Java GUI design and development, reference can be made to the following technical publications: "JAVA" (1997) by Ed Tittel and Bill Brogden, published by IDG Books Worldwide, Inc.; "Wilde's WWW: Technical Foundations of the World Wide Web" (1999) by Erik Wilde, published by Springer-Verlag, Berlin, Heidelberg; "The Java™ Tutorial Second Edition: Object Oriented Programming For The Internet" (1999), by Mary Campione and Kathy Walrath, published by Sun Microsystems, Inc., of Palo Alto, Calif.; and "The JFC Swing Tutorial: A Java Guide To Constructing JAVA GUIs" (1999), by Mary Campione and Kathy Walrath, published by Sun Microsystems, Inc., of Palo Alto, Calif.; each said publication being incorporated herein by reference.

[0453] After writing/authoring the source code for the Applet, the CPIR-enabling Java Applet is ascribed a unique name such as, for example, "UPNXXXXXXYYYYYZ" for a 12 digit Uniform Product Code.

[0454] As indicated at Block A2 in **FIG. 4G1**, the second step of the method involves compiling the source code of the Applet into Java bytecode, and then placing/loading the classfiles for the Applet within the server_root/Applets directory on the Java Web Server **11**".

[0455] As indicated at Block B1 in **FIG. 4G1**, the third step of the method involves for each UPN-specified consumer product, (1) containing the complete Applet HTML tag <APPLET> within an executable file, and (2) storing each such Applet tag containing file in the Central CPIR-Enabling Applet Library on the RDBMS server **9**, as shown in **FIG. 4H1**.

[0456] As indicated at Block C in **FIG. 4G1**, the fourth step of the method involves distributing the CPIR-enabling

Applet HTML tags to retailers, wholesalers, advertisers, and others who desire to deliver UPN-directed CPI search results to their customers, clients and the like. This distribution process may be carried out in a number of ways.

[0457] For example, in one embodiment of the present invention shown in FIG. **4M1**, retailers, wholesalers, advertisers, and others would visit the IPI Central WWW site as shown in **FIG. 3C** and selected the Applet Tag Download/Distribute Mode by selecting mode control button **21F**, whereupon a (Java) GUI, is displayed in the display frame **20C**. Within this GUI, a "master" list of executable files containing CPIR-enabling Applet tags is displayed for each consumer product registered within the system. The CPIR-enabling Applet tag containing file associated with any particular consumer product can be searched for by UPN, trademark, and/or product descriptor, and once found, can be viewed and simply downloaded to a remote client computer system connected to the Internet using, for example, ftp or other electronic data or document interchange protocols (e.g. XML/ICE) suitable for carrying out the <APPLET> tag transport process. Downloaded CPIR-enabling Applet tag containing files can be then stored in a local CPIR-enabling Applet/Servlet Library maintained on a client computer until it is time to extract the Applet tag therefrom and embed the same into a particular HTML-encoded document.

[0458] In an alternative embodiment shown in **FIG. 4M2**, the distribution of CPIR-enabling Applet tags is enabled by providing a "CPIR-enabling Applet Tag Download" Link/Button **100** on the Web page of each product being offered for sale in an EC-enabled store or product catalog maintained by a manufacturer, retailer or other party. This inventive feature should be most useful in BC-enabled Business-to-Business (i.e. Vendor-to-Retailer) UPC-based Product Catalogs such as, for example, QRS's Keystone UPC Product Catalog, where purchasing agents of retailers could download "Applet tag containing" files, along with product images and other product information after placing a purchase order therethrough, for use in constructing the retailer's EC-enabled (retailer-to-consumer) store or product catalog. Using this method of the present invention, consumers would be provided with instant manufacturer-defined product information prior to, and/or after a consumer purchase at an EC-enable store on the WWW, thereby greatly improving the consumer shopping experience on the WWW.

[0459] As indicated at Block D in **FIG. 4G2**, the fifth step of the method involves enabling retailers, wholesalers, advertisers, and others to (1) open the downloaded Applet tag containing files, (2) extract the CPIR-enabling HTML tags contained therewithin, and (3) embed (i.e. insert) one or more distributed CPIR-enabled Applet tags into acceptable HTML-encoded documents associated with EC-enabled WWW sites, EC-enabled storefronts and catalogs, Internet product advertisements, on-line auction-based WWW sites, or other types of Web-documents.

[0460] In general, this step of the method involves first creating or otherwise procuring a suitable HTML-encoded document which may understandably include other types of code (e.g. XML) therein, other than HTML code. While such HTML documents can be created using any HTML-editing program, such as BBD-Edit, it is expected that in most applications the underlying HTML-encoded document will be generated using tools such as, for example: GO-LIVE® WWW-Site Development and Management solution software from Adobe Systems, Inc. to create the HTML pages associated with a particular WWW site; Catalog-Maker™ and CatalogManager electronic commerce solution software programs from RealEDI, Inc; Intershop 4 Enfinity™ Electronic Commerce Solution software from Intershop Communications, Inc; and/or any other commercially available HTML-authoring tools which enable quick and easy creation of HTML-encoded documents, and easy insertion of any downloaded CPIR-enabling Applet HTML tag using, for example, simple commands or drag-and-drop procedures.

[0461] As indicated at Block E in **FIG. 4G2**, the sixth step of the method involves serving HTML documents with inserted CPIR-enabling Applet tags, from Internet information servers to Java-enabled client computer subsystems **13** operated by consumers at home, in the office, in EC-enabled and "brick and mortar" retail stores, or on the road, as the case may be. As shown in **FIG. 4H1**, such Internet information servers can include, for example, IPI servers **12**, retailer-related EC-enabled information servers **12A**, manufacturer-related EC-enabled information servers **12B**, and/or any other Internet (http or ftp) information servers operating on the Internet from which HTML-encoded document are served for any informational, educational, and/or entertainment purpose.

[0462] As indicated at Block F in **FIG. 4G2**, the seventh step of the method hereof involves using a Java-enabled client computer subsystem **13** to display served HTML-encoded documents having one or more of CPIR-enabling Applet tags embedded therewithin. This step is carried out by the consumer pointing his or her Java-enabled browser program (e.g. Netscape Navigator, Microsoft Explorer, or Sun Microsystems' HotJava program) to an HTML-encoded document within which a CPIR-enabling Java Applet tag is embedded, at a particular point of presence on the WWW.

[0463] As shown in FIGS. **4M1** through **4R2**, CPIR-enabling Applets can be graphically-encoded in an variety of different ways to provide the consumer with a visual indication that, clicking on the graphical object, however manifested, will automatically result in a CPI search on a particular product identified by a UPN encoded within the associated Applet.

[0464] In the illustrative embodiments shown in FIGS. **4P1**, **4Q1**, **4R1** and **4S1**, servicemarks such as "UPC Request™ Cyberservice™ URL Search" serve to inform the consumer that the object, if selected from the displayed Web page, will cause a URL search to be performed with respect to the particular consumer product and the results thereof displayed the "point of presence" of the consumer which may be at a particular point in an EC-enabled store (e.g. at the check-out display screen or POS), at on-line auction site, at a Web-based product advertisement, or anywhere else on the WWW. Notably, an important advantage provided by this information search technique of the present invention is that it does not disturb the consumer at his or her point of presence (or sale), where ever that may be. Instead, the CPI-based search and display method of the present invention enables the delivery of accurate product-specific manufacturer-defined information at a particular point in Cyberspace by the consumer performing a single mouse-clicking operation. This enables the consumer to make an informed

decision thereat based on the information displayed in the corresponding Java GUI generated upon launching the CPIR-enabling Applet at the consumer's point of presence on the WWW.

[0465] It is understood, however, that other techniques may be used to create a visual indication to the consumer that a CPIR-enabling Applet is located at a particular point on the WWW and that if this Applet is executed (e.g. by a single mouse-clicking operation), then a UPN-directed consumer product information search will be automatically executed and the results therefrom will be displayed within a Java GUI at that point of presence. One alternative technique would be to embed the CPIR-enabling Applet within a thumb-nail or large size photo-image of the consumer product being offered for sale, lease, auction, or other purpose on the WWW. Using this technique, the consumer need only click on the image to initiate a UPN-directed consumer product information search on the IPI Registrant Database (e.g. RDBMS Server **9**) of the system.

[0466] Notably, the person or persons responsible for delivering product advertisements to particular locations on one or more WWW sites can use the OPEN ADSTREAM™ (OAS) 5.0 Internet Advertisement Management Solution software from Real Media, Inc., of New York, N.Y., and any other suitable software solution, running on the Internet (http) information server (**12**, **12'**, **12A** or **12B**), and managed using an Web-enabled client subsystem **13**, as shown in **FIG. 4H2**. Using the OAS 5.0 advertisement management solution, and the CPI search and display method of the present invention described above, a webmaster or advertising manager assigned to a particular Internet information server (**12**, **12'**, **12A** or **12B**) can: (1) access the Web-based product advertisement for a particular product (i.e. HTML code, image files, and any other rich media content associated therewith); (2) access previously downloaded CPIR-enabling Java Applet(s) for the consumer product, stored in a locally-maintained "Library (i.e. Catalog) of CPIR-Enabling Applets" on a client machine or server on the network, or directly access CPIR-enabling Java Applets from the centrally-maintained "Library (i.e. Catalog) of CPIR-Enabling Applets" illustrated in **FIG. 4H2**; and (3) use Real Media's OAS 5.0 solution software to deliver both the Web-based product advertisement (i.e. its HTML code, and other media-rich content) and the corresponding CPIR-enabling Applet tag (and image files associated therewith) to a designated section on a particular Web page of a specific WWW-site.

[0467] It is understood that there are different ways of inserting/embedding both of these objects within a particular section of an HTML-encoded document using the highly-advanced HTML-editing functionalities of the OAS 5.0 software system. For example, the Web-based product advertisement can be inserted within a first spatially-defined portion of the target HTML document (occupying the largest portion of the purchased Internet advertising space), while the CPIR-enabling client-side Applet is inserted within a second spatially-defined portion of the target HTML document occupying the balance of the purchased Internet advertising space. Alternatively, both the CPIR-enabling Applet and the Web-based product advertisement can be inserted within substantially same spatially-defined portion of the target HTML document so as to achieve spatial overlap therebetween. This way when the consumer clicks on the advertisement image, or some preselected portion thereof, the underlying CPIR-enabling Applet will be automatically executed and the corresponding Java GUI generated for displaying the results of the UPN-directed database search.

[0468] In situations where the Internet product advertisement (e.g. banner advertisement) embodies a Java Applet tag which, when executed, produces a new Java GUI (i.e. new browser interface), then a CPIR-enabling Applet tag can be embedded within the HTML-encoded document displayed in the new Java GUI.

[0469] Other ways of embedding the Web-based advertisement and the related CPIR-enabling Applet tags will become apparent hereinafter to those skilled in the art having had the benefit of reading the present disclosure.

[0470] As indicated at Block G in **FIG. 4G2**, the eight step in the method involves the consumer recognizing that a CPIR-enabling Applet tag is embedded within a Web-document displayed on a Java-enabled client computer subsystem **13**, and thereafter launching/executing the associated Applet to initiate a UPN-directed search within the RDBMS server **9** by performing a single mouse clicking operation.

[0471] Notably, the second illustrative embodiment described above has been described with particular focus given to CPIR-enabling Applets encoded with the UPN of a particular consumer product. It is understood, however, that the CPIR-enabling Applets of the present invention can be encoded with the trademark(s) used in connection with a particular consumer product, thus providing Trademark-encoded CPIR-enabling Applets, in contrast with UPN-encoded CPIR-enabling Applets. In such alternative embodiments, the encoded trademark would be used to direct a search through the RDBMS server **9**, and display the results thereof in a new (independent) Java GUI generated at the point of Applet tag embodiment. Alternatively, a product descriptor associated with a particular product can be encoded within the corresponding CPIR-enabling Applet, used to direct a search through the RDBMS server **9**, and display the results thereof in an independent Java GUI generated at the point of Applet tag embodiment.

The Third Applet-Driven Method of Accessing and Displaying Categorized UPN/URL Link Menus From the UPN/URL Database Management Subsystem the Present Invention

[0472] After providing a brief overview on the system architecture of **FIG. 2B3** and the nature of the client-side CPIR-enabling Applets deployed therewithin, the steps associated with Applet-driven CPI-acquisition method of the third illustrative embodiment will be described in detail with reference to FIGS. **4I1**, **4I2**, **4J1** and **4J2**.

[0473] In general, the method of FIGS. **4I1** and **4I2**, like that of FIGS. **4G1** and **44G2**, involves using a client-side CPIR-enabling Applet to automatically conduct a UPN-directed search on the UPN/URL Database Management Subsystem hereof (i.e. RDBMS server **9**) in response to a single mouse-clicking operation by the consumer on the HTML tag associated with the CPIR-enabling Applet. In the illustrative embodiment, the CPIR-enabling Applet of the present invention is a program written in the Java™ programming language and has an HTML tag (indicated by <APPLET>) which is designed to be included in an HTML

page, much in the same way an image can be included therewithin (in accordance with the HTML 3.2 Specification).

[0474] In the method of the second illustrative embodiment, CPIR-enabling Applets are designed to work within a request/response processing model, as shown in **FIG. 2B3**. In this request/response model, a client subsystem **13** sends a request message to the Java Web Server **11'"** and the server **11'"** responds by sending back a reply message. In the illustrative embodiment, requests come in the form of http, although it is understood that other protocols (e.g. ftp, EDI or a custom protocol) may be used. The request and the corresponding response reflect the state of the client and the server at the time of the request.

[0475] When using a Java-enabled browser to view a Web page containing a client-side CPIR-enabling Applet tag <APPLET>, the Applet's compiled class code is automatically accessed from the Java Web Server **11'"** and executed on the client-side of the network connection illustrated in **FIG. 2B3**. Thus, as shown in **FIG. 2B3**, the Java-enabled client machine **13** in this network architecture must run a Java-enabled browser program that provides a Java Virtual Machine (JVM) for running/executing Java Applets on the client-side thereof, in much the same way that Java Web Server **11'** must provide a JVM for running Java servlets on the server side thereof, as shown in **FIG. 2B1**.

[0476] With reference to **FIG. 2B3**, the method of creating, loading, distributing, embedding and executing client-side CPIR-enabling Java Applets in accordance with the principles of the present invention schematically illustrated in FIGS. **4I1, 4I2, 4J1** and **4J2** will now be described below.

[0477] As indicated at Block A1 in **FIG. 4I1**, the first step of the method involves using the Java Applet API to write or otherwise author the source code for a client-side CPIR-enabling Java Applet for each UPN-specified consumer product registered in the RDBMS server **9**. In general, the source code for each client-side CPIR-enabling Java Applet (i.e. Applet) will embody one or more of following items of information: (i) the UPN of the particular product on which the CPI search is to be carried out and the search results thereof displayed; (ii) Java classes required for performing a UPN-directed search on the RDBMS server **9** using a "socket connection" between the Java-enabled client subsystem **13** and the Java Web Server **11'"**, producing a particular Java GUI for displaying the results obtained from the UPN-directed search; and (iii) license-related information specifying the terms and conditions of the CPIR-enabling Applet license and conditions under which the CPIR-enabling Applet shall operate.

[0478] Notably, such license-related information may specify: (1) one or more specific host domains (e.g. www.homedepot.com or www.walmart.com) from which a Web document containing the corresponding Applet tag may launch the CPIR-enabling Applet under a licensing program; (2) one or more general Internet domains (e.g. .com, .org., .gov, .int, .mil, .uk, etc.) from which a Web document containing the corresponding Applet tag may launch the CPIR-enabling Applet under a licensing program; (3) the time duration of the licensing period associated with the CPIR-enabling Applet; and (4) any other restrictions set by the associated manufacturer and/or retailer, and/or administrator of the consumer product information system of the present invention, that must be satisfied for a registered CPIR-enabled Applet to operate within a Web-document.

[0479] Notably, the Java source code for each CPIR-enabling Applet will vary depending upon implementation. However, regardless of the particular implementation, it can be expected that each CPIR-enabling Applet designed, for example, to search an (Oracle-based JDBC) RDBMS Server **9** for current UPC/URL list symbolically linked to a specified UPN and thereafter display the results in an independent Java GUI, will typically include Java source code specifying:

[0480] (1) the importable JDBC classes required by the CPIR-enabling Applet;

[0481] (2) the importable java classes to be used in the CPIR-enabling Applet;

[0482] (3) the JDBC driver to be loaded for the Oracle-based RDBMS server **9**;

[0483] (4) the connection strings to the RDBMS server **9**;

[0484] (5) the CPI query to be executed on the UPN/URL Database, dependent on the UPN of the associated consumer product and possibly other search criteria and Applet licensing conditions;

[0485] (6) the Applet tag, its graphical icon or alias to trigger execution the Applet and its associated CPI query;

[0486] (7) the CPI Search Result GUI to be displayed on requesting client and its relative location to the associated Applet tag; and

[0487] (8) the operations that will be carried out upon execution of the CPI query including

[0488] Boolean search logic to be carried out upon initiation of the UPN-directed CPI search;

[0489] if a new connection is required between Java Web Server **11'"** and the RDBMS server **9**;

[0490] Loading the JDBC driver;

[0491] Connecting to the RDBMS server **9**;

[0492] Creating a SQL statement based on the specified Boolean search logic and UPN;

[0493] Executing the SQL query statement; and

[0494] Dumping the search results to the CPI Search Result GUI.

[0495] When using earlier versions of the HTML Specification (i.e. HTML 3.2 by the World Wide Web Consortium), the source code for each CPIR-enabling Java Applet will adhere to the general syntax of that the HTML 3.2 Specification. Also, if the HTML 4.0 Specification is used, then the source code for each CPIR-enabling Java Applet will adhere to the general syntax of the HTML 4.0 Specification, as discussed above.

[0496] After writing/authoring the source code for the CPIR-enabling Java Applet, the Applet is ascribed a unique name such as, for example, "UPNXXXXXXYYYYYZ" for a 12 digit Uniform Product Code.

[0497] As indicated at Block A2 in **FIG. 4I1**, the second step of the method involves compiling the source code of the Applet into Java bytecode, and then placing/loading the classfiles for the Applet within the server_root/Applets directory on the Java Web Server **11'''**.

[0498] As indicated at Block B1 in **FIG. 4I1**, the third step of the method involves for each UPN-specified consumer product, (1) containing the complete Applet HTML tag <APPLET> within an executable file, and (2) storing each such Applet tag containing file in the Central CPIR-Enabling Applet Library on the RDBMS server **9**, as shown in **FIG. 4J1**.

[0499] As indicated at Block C in **FIG. 4I1**, the fourth step of the method involves distributing the CPIR-enabling Applet HTML tags to retailers, wholesalers, advertisers, and others who desire to deliver UPN-directed CPI search results to their customers, clients and the like. This distribution process may be carried out in several different ways which have been detailed hereinabove in connection with the second illustrative method illustrated in FIGS. **4G1** through **4H2** and described above.

[0500] As indicated at Block D in **FIG. 4G2**, the fifth step of the method involves enabling retailers, wholesalers, advertisers, and others to (1) open the downloaded Applet tag containing files, (2) extract the CPIR-enabling HTML tags contained therewithin, and (3) embed (i.e. insert) one or more distributed CPIR-enabled Applet tags into acceptable HTML-encoded documents associated with EC-enabled WWW sites, EC-enabled storefronts and catalogs, Internet product advertisements, on-line auction-based WWW sites, or other types of Web-documents.

[0501] In general, this step of the method involves first creating or otherwise procuring a suitable HTML-encoded document which may understandably include other types of code (e.g. XML) therein, other than HTML code. While such HTML documents can be created using any HTML-editing program, such as BBD-Edit, it is expected that in most applications the underlying HTML-encoded document will be generated using tools such as, for example: GO-LIVE® WWW-Site Development and Management solution software from Adobe Systems, Inc. to create the HTML pages associated with a particular WWW site; Catalog-Maker™ and CatalogManager electronic commerce solution software programs from RealEDI, Inc; Intershop 4 Enfinity™ Electronic Commerce Solution software from Intershop Communications, Inc; and/or any other commercially available HTML-authoring tools which enable quick and easy creation of HTML-encoded documents, and easy insertion of any down-loaded CPIR-enabling Applet HTML tag using, for example, simple commands or drag-and-drop procedures.

[0502] As indicated at Block E in **FIG. 4G2**, the sixth step of the method involves serving servlet tag encoded HTML documents from Internet information servers to Java-enabled client computer subsystems **13** operated by consumers at home, in the office, in EC-enabled and "brick and mortar" retail stores, or on the road, as the case may be. As shown in **FIG. 4H1**, such Internet information servers can include, for example, IPI servers **12**, retailer-related EC-enabled information servers **12A**, manufacturer-related EC-enabled information servers **12B**, and/or any other Internet (http or ftp) information servers operating on the Internet from which HTML-encoded document are served for any informational, educational, and/or entertainment purpose.

[0503] As indicated at Block F in **FIG. 4G2**, the seventh step of the method hereof involves using a Java-enabled client computer subsystem **13** to display served HTML-encoded documents having one or more of CPIR-enabling Applet tags embedded therewithin. This step is carried out by the consumer pointing his or her Java-enabled browser program (e.g. Netscape Navigator, Microsoft Explorer, or Sun Microsystems' HotJava program) to an HTML-encoded document within which a CPIR-enabling Java Applet tag is embedded, at a particular point of presence on the WWW. As shown in FIGS. **4M1** through **4R2**), CPIR-enabling Applets can be graphically-encoded in an variety of different ways as described in detail detailed hereinabove in connection with the second illustrative method illustrated in FIGS. **4G1** through **4H2** and described above.

[0504] As indicated at Block G in **FIG. 4G2**, the eight step in the method involves the consumer recognizing that a CPIR-enabling Applet tag is embedded within a Web-document displayed on a Java-enabled client computer subsystem, and thereafter launching/executing the associated Applet to initiate a UPN-directed search within the RDBMS server **9** by performing a single mouse clicking operation.

[0505] Notably, the third illustrative embodiment has been described with particular focus given to CPIR-enabling Applets encoded with the UPN of a particular consumer product. It is understood, however, that the CPIR-enabling Applets of the present invention can be encoded with the trademark(s) used in connection with a particular consumer product, thus providing Trademark-encoded CPIR-enabling Applets, in contrast with UPN-encoded CPIR-enabling Applets. In such alternative embodiments, the encoded trademark would be used to direct a search through the RDBMS server **9**, and display the results thereof in a new (independent) Java GUI generated at the point of Applet tag embodiment. Alternatively, a product descriptor associated with a particular product can be encoded within the corresponding CPIR-enabling Applet, used to direct a search through the RDBMS server **9**, and display the results thereof in an independent Java GUI generated at the point of Applet tag embodiment.

The Fourth Applet-Driven Method of Accessing and Displaying Categorized UPN/URL Link Menus From the UPN/URL Database Management Subsystem the Present Invention

[0506] After providing a brief overview on the system architecture of **FIG. 2B4** and the nature of the client-side CPIR-enabling Applets deployed therewithin, the steps associated with Applet-driven CPI-acquisition method of the fourth illustrative embodiment will be described in detail with reference to FIGS. **4K1**, **4K2**, **4L1** and **4L2**.

[0507] In general, the method of FIGS. **4K1** and **4IK2**, like that of FIGS. **4G1** and **4G2** and **411** and **412** involves using a client-side CPIR-enabling Applet to automatically conduct a UPN-directed search on the UPN/URL Database Management Subsystem hereof (i.e. RDBMS server **9**) in response to a single mouse-clicking operation by the consumer on the HTML tag associated with the CPIR-enabling Applet. In the illustrative embodiment, the CPIR-enabling Applet of the present invention is a program written in the Java™ pro-

gramming language and has an HTML tag (indicated by <APPLET>) which is designed to be included in an HTML page, much in the same way an image can be included therewithin (in accordance with the HTML 3.2 Specification).

[0508] In the method of the second illustrative embodiment, CPIR-enabling Applets are designed to work within a request/response processing model, as shown in **FIG. 2B4**. In this request/response model, a client subsystem **13** sends a request message to the Java Web Server **11**"" and the server **11**"" responds by sending back a reply message. In the illustrative embodiment, requests come in the form of http, although it is understood that other protocols (e.g. ftp, EDI or a custom protocol) may be used. The request and the corresponding response reflect the state of the client and the server at the time of the request.

[0509] When using a Java-enabled browser to view a Web page containing a client-side CPIR-enabling Applet tag <APPLET>, the Applet's compiled class code is automatically accessed from the Java Web Server **11**"" and executed on the client-side of the network connection illustrated in **FIG. 2B4**. Thus, as shown in **FIG. 2B3**, the Java-enabled client machine **13** in this network architecture must run a Java-enabled browser program that provides a Java Virtual Machine (JVM) for running/executing Java Applets on the client-side thereof, in much the same way that Java Web Server **11'** must provide a JVM for running Java servlets on the server side thereof, as shown in **FIG. 2B1**.

[0510] With reference to **FIG. 2B4**, the method of creating, loading, distributing, embedding and executing client-side CPIR-enabling Java Applets in accordance with the principles of the present invention schematically illustrated in FIGS. **4K1**, **4K2**, **4L1** and **4L2** will now be described below.

[0511] As indicated at Block **A1** in **FIG. 4K1**, the first step of the method involves using the Java Applet API to write or otherwise author the source code for a client-side CPIR-enabling Java Applet for each UPN-specified consumer product registered in the RDBMS server **9**. In general, the source code for each client-side CPIR-enabling Java Applet (i.e. Applet) will embody one or more of following items of information: (i) the UPN of the particular product on which the CPI search is to be carried out and the search results thereof displayed; (ii) Java classes required for performing a UPN-directed search on the RDBMS server **9** using a Remote Invocation Method (RMI) executed on the Java Web Server **11**"", producing a particular Java GUI for displaying the results obtained from the UPN-directed search; and (iii) license-related information specifying the terms and conditions of the CPIR-enabling Applet license and conditions under which the CPIR-enabling Applet shall operate.

[0512] Notably, such license-related information may specify: (1) one or more specific host domains (e.g. www.homedepot.com or www.walmart.com) from which a Web document containing the corresponding Applet tag may launch the CPIR-enabling Applet under a licensing program; (2) one or more general Internet domains (e.g. .com, .org., .gov, .int, .mil, .uk, etc.) from which a Web document containing the corresponding Applet tag may launch the CPIR-enabling Applet under a licensing program; (3) the time duration of the licensing period associated with the CPIR-enabling Applet; and (4) any other restrictions set by the associated manufacturer and/or retailer, and/or administrator of the consumer product information system of the present invention, that must be satisfied for a registered CPIR-enabled Applet to operate within a Web-document.

[0513] The RMI on Java Web Server **11**"" enables connectivity between Java Web Server **11**"". and the RDBMS Server **9** using the standard Java native method interface (JNI) or the standard JDBC package. At its most basic level, RMI is Java's remote procedure call (RPC) mechanism enabling connectivity to the RDBMS server **9** using native methods. Further details on the RMI are published in the Technical Paper "Java Remote Method Invocation -Distributed Computing For Java" by JavaSoft, set forth at http://www.javasoft.com/marketing/collateral/javarmi.html, incorporated herein by reference.

[0514] Notably, the Java source code for each CPIR-enabling Applet will vary depending upon implementation. However, regardless of the particular implementation, it can be expected that each CPIR-enabling Applet designed, for example, to search an (Oracle-based JDBC) RDBMS Server **9** for current UPC/URL list symbolically linked to a specified UPN and thereafter display the results in an independent Java GUI, will typically include Java source code specifying:

[0515] (1) the importable JDBC classes required by the CPIR-enabling Applet;

[0516] (2) the importable java classes to be used in the CPIR-enabling Applet;

[0517] (3) the JDBC driver to be loaded for the Oracle-based RDBMS server **9**;

[0518] (4) the connection strings to the RDBMS server **9**;

[0519] (5) the CPI query to be executed on the UPN/URL Database, dependent on the UPN of the associated consumer product and possibly other search criteria and Applet licensing conditions;

[0520] (6) the Applet tag, its graphical icon or alias to trigger execution the Applet and its associated CPI query;

[0521] (7) the CPI Search Result GUI to be displayed on requesting client and its relative location to the associated Applet tag; and

[0522] (8) the operations that will be carried out upon execution of the CPI query including

[0523] Boolean search logic to be carried out upon initiation of the UPN-directed CPI search;

[0524] if a new connection is required between Java Web Server **11**"" and the RDBMS server **9**;

[0525] Loading the JDBC driver;

[0526] Connecting to the RDBMS server **9**;

[0527] Creating a SQL statement based on the specified Boolean search logic and UPN;

[0528] Executing the SQL query statement; and

[0529] Dumping the search results to the CPI Search Result GUI.

[0530] When using earlier versions of the HTML Specification (i.e. HTML 3.2 by the World Wide Web Consortium), the source code for each CPIR-enabling Java Applet will adhere to the general syntax of that the HTML 3.2 Specification. Also, if the HTML 4.0 Specification is used, then the source code for each CPIR-enabling Java Applet will adhere to the general syntax of the HTML 4.0 Specification, as discussed above.

[0531] After writing/authoring the source code for the CPIR-enabling Java Applet, the Applet is ascribed a unique name such as, for example, "UPNXXXXXXYYYYYZ" for a 12 digit Uniform Product Code.

[0532] As indicated at Block A2 in **FIG. 4K1**, the second step of the method involves compiling the source code of the Applet into Java bytecode, and then placing/loading the classfiles for the Applet within the server_root/Applets directory on the Java Web Server 11"".

[0533] As indicated at Block B1 in **FIG. 4K1**, the third step of the method involves for each UPN-specified consumer product, (1) containing the complete Applet HTML tag <APPLET> within an executable file, and (2) storing each such Applet tag containing file in the Central CPIR-Enabling Applet Library on the RDBMS server **9**, as shown in **FIG. 4J1**.

[0534] As indicated at Block C in **FIG. 411**, the fourth step of the method involves distributing the CPIR-enabling Applet HTML tags to retailers, wholesalers, advertisers, and others who desire to deliver UPN-directed CPI search results to their customers, clients and the like. This distribution process may be carried out in several different ways which have been detailed hereinabove in connection with the second illustrative method illustrated in FIGS. **4G1** through **4H2** and described above.

[0535] As indicated at Block D in **FIG. 4K2**, the fifth step of the method involves enabling retailers, wholesalers, advertisers, and others to (1) open the downloaded Applet tag containing files, (2) extract the CPIR-enabling HTML tags contained therewithin, and (3) embed (i.e. insert) one or more distributed CPIR-enabled Applet tags into acceptable HTML-encoded documents associated with EC-enabled WWW sites, EC-enabled storefronts and catalogs, Internet product advertisements, on-line auction-based WWW sites, or other types of Web-documents.

[0536] In general, this step of the method involves first creating or otherwise procuring a suitable HTML-encoded document which may understandably include other types of code (e.g. XML) therein, other than HTML code. While such HTML documents can be created using any HTML-editing program, such as BBD-Edit, it is expected that in most applications the underlying HTML-encoded document will be generated using tools such as, for example: GO-LIVE® WWW-Site Development and Management solution software from Adobe Systems, Inc. to create the HTML pages associated with a particular WWW site; Catalog-Maker™ and CatalogManager electronic commerce solution software programs from RealEDI, Inc; Intershop **4** Enfinity™ Electronic Commerce Solution software from Intershop Communications, Inc; and/or any other commercially available HTML-authoring tools which enable quick and easy creation of HTML-encoded documents, and easy insertion of any downloaded CPIR-enabling Applet HTML tag using, for example, simple commands or drag-and-drop procedures.

[0537] As indicated at Block E in **FIG. 4K2**, the sixth step of the method involves serving servlet tag encoded HTML documents from Internet information servers to Java-enabled client computer subsystems **13** operated by consumers at home, in the office, in EC-enabled and "brick and mortar" retail stores, or on the road, as the case may be. As shown in **FIG. 4H1**, such Internet information servers can include, for example, IPI servers **12**, retailer-related EC-enabled information servers **12A**, manufacturer-related EC-enabled information servers **12B**, and/or any other Internet (http or ftp) information servers operating on the Internet from which HTML-encoded document are served for any informational, educational, and/or entertainment purpose.

[0538] As indicated at Block F in **FIG. 4G2**, the seventh step of the method hereof involves using a Java-enabled client computer subsystem **13** to display served HTML-encoded documents having one or more of CPIR-enabling Applet tags embedded therewithin. This step is carried out by the consumer pointing his or her Java-enabled browser program (e.g. Netscape's Navigator, Microsoft's Internet Explorer, or Sun Microsystems' HotJava program) to an HTML-encoded document within which a CPIR-enabling Java Applet tag is embedded, at a particular point of presence on the WWW. As shown in FIGS. **4M1** through **4R2**, CPIR-enabling Applets can be graphically-encoded in an variety of different ways as described in detail detailed hereinabove in connection with the second illustrative method illustrated in FIGS. **4G1** through **4H2** and described above.

[0539] As indicated at Block G in **FIG. 4G2**, the eight step in the method involves the consumer recognizing that a CPIR-enabling Applet tag is embedded within a Web-document displayed on a Java-enabled client computer subsystem, and thereafter launching/executing the associated Applet to initiate a UPN-directed search within the RDBMS server **9** by performing a single mouse clicking operation.

[0540] Notably, the third illustrative embodiment has been described with particular focus given to CPIR-enabling Applets encoded with the UPN of a particular consumer product. It is understood, however, that the CPIR-enabling Applets of the present invention can be encoded with the trademark(s) used in connection with a particular consumer product, thus providing Trademark-encoded CPIR-enabling Applets, in contrast with UPN-encoded CPIR-enabling Applets. In such alternative embodiments, the encoded trademark would be used to direct a search through the RDBMS server **9**, and display the results thereof in a new (independent) Java GUI generated at the point of Applet tag embodiment. Alternatively, a product descriptor associated with a particular product can be encoded within the corresponding CPIR-enabling Applet, used to direct a search through the RDBMS server **9**, and, display the results thereof in an independent Java GUI generated at the point of Applet tag embodiment.

[0541] While the illustrative embodiments described above have employed Java Applet technology, which is designed to work with nearly all modern Internet browser programs, it is understood, however, that it is possible to use Active-X type objects (i.e. Active-X Applets) embedded within Web-documents, such as XML and SGML encoded documents including Active Server Pages (ASPs) from the Microsoft Corporation, in order to implement UPN-directed

methods or the present invention at the point of presence of the consumer within a Cyberspace environment. Such alternative embodiments are a straightforward application of the techniques and technology disclosed hereinabove and thus fall within the scope and spirit of the present invention.

[0542] Also, while the above-described method of information searching, access and display has been described in connection with consumer products, it is understood that the principles of the present invention can also be used to deliver Web-based information to consumers in connection with a particular consumer service which has been assigned a Universal Service Number (USN) that functions in a similar manner to a UPN used in connection with a particular consumer product. In such alternative embodiments, the UPN/URL Database Management Subsystem **9** can be readily extended to contain symbolic links between Universal Service Numbers (USN) and URLs to form a UPS/URL database along the principles described hereinabove.

[0543] Also, the CPIR-enabling Applets of the present invention may be modified to provide consumers with general access to any IPI WWW site in accordance with the present invention, and not necessarily a product-specific Cyber-Service™ search, as described above. Thus, for example, in the case where the CPIR-enabling Applet is not encoded with any particular UPN, then the CPIR-enabling Applet will generate and display a pop-up Java GUI at the point where its Applet tag (or related image IMG) is embedded. Such as Java GUI could be designed to enable either (1) a generalized (unrestricted) consumer product information display, as would be desired at WWW search Engines/Directories such as Yahoo, Lycos, Excite, Alta-Vista, and the like, or (2) a restricted consumer product information display, as would be desired by a particular retailer operating an EC-enabled store or on-line catalog where browsing for merchandise not carried in the store or catalog is not to be encouraged.

[0544] An example of a CPIR-enabling Applet designed to produce a Java GUI for the "manufacturer-unrestricted or generalized" UPC Request CPI Service is illustrated in FIGS. **4N1** and **4Q1** by using a graphical icon or button, displayed on the lower portion of each display screen, and labeled as "UPC REQUEST™ CENTRAL Product Information Search". An example of the Java GUIs produced by these CPIR-enabling Applets are illustrated in **FIG. 4N2** and **4Q2**, respectively.

[0545] An example of a CPIR-enabling Applet designed to produce a CPID-enabling Java GUI for the "manufacturer-restricted" UPC Request Retailer CPI Service is indicated in **FIG. 401** by a graphical icon or button, displayed on the lower portion of each display screen, and labeled as "UPC REQUEST™ Retail Product Information @ SPORTS PLACE'. An example of the Java GUI produced by this CPIR-enabling Applet is indicated in **FIG. 402**. Notably, this type of CPIR-enabling Applet provides consumers with desired information about the UPN-encoded product, while disabling the consumer from browsing for merchandise not carried in the EC-oriented store or catalog of the hosting retailer.

[0546] As illustrated above, in the case where the CPIR-enabling Applet is encoded with a particular UPN, then the function of the CPIR-enabling Applet will be to generate and display an independent pop-up Java GUI at the point where the Applet tag (or associated image) is embedded, for displaying the search results made against the consumer product identified by the UPN embodied within the CPIR-enabling Applet. An example of a CPIR-enabling Applet designed to produce a CPID-enabling Java GUI for a Cyber-Service URL Search is indicated in FIGS. **4PI** and **4RI** by a graphical icon or button, displayed on the lower portion of each display screen, and labeled as "UPC REQUEST™ Cyber-Service™ URL Search." Notably, operation of this type of CPIR-enabling Applet can be restricted to a particular retailer (or manufacturer) by the inclusion of a domain name constraint within the Applet itself, as described hereinabove. In the case of the Cyber-Service URL Search of the UPC Request System, the executed CPIR-enabling Applet automatically returns for display a menu of categorized URLs symbolically linked to the encoded UPN by the manufacturer and/or its agent. It would be desirable to embed this type of CPIR-enabling Applet on Web-documents in an EC-enabled stores and on-line catalogs of a particular retailer or manufacturer, displaying consumer products to be purchased, as well as on Web-documents serving as Internet-based product advertisements.

[0547] Referring to FIGS. **4N1** and **4N2**, the above-described method of CPI searching and display will now be illustrated in the context of browsing a WWW Search Directory or Engine, and looking for a simple yet effective way of finding accurate consumer product related information on a particular product, or class of products. When searching for consumer product information at a WWW Search Directory or Engine, such as Yahoo, Excite, Alta Vista, Lycos, etc., it will be desirable for the consumer to search against all manufacturers within the entire UPN/URL Database Management Subsystem **9** before returning the search results to the consumer for display. Therefore, in this sort of Cyberspace environment, it will be oftentimes desirable to embed a CPIR-enabling Applet in the home-page of the WWW search directory or engine so that, upon clicking the graphical icon thereof, an independent Java GUI to the UPC Request Central WWW site will be automatically produced so that all modes of searching are made available to the consumer against all manufacturers registered (and possibly unregistered) within the UPN/URL Database Management Subsystem **9**, as shown in **FIG. 4N2**. Notably, this Java GUI is very similar to the Java GUI set forth in **FIG. 3C**.

[0548] Referring to FIGS. **4O1** through **4O2**, the above-described method of CPI searching and display is illustrated in a different context, wherein a consumer is shopping/browsing an EC-enabled storefront of a particular retailer, and considering whether or not to make an on-line purchase of a particular consumer product displayed within the catalog pages thereof. In this sort of environment, the retailer will typically prefer that the consumer can only search on manufacturers of merchandise being offered for sale within the EC-enabled store, lest the consumer will encouraged to leave upon finding out that what he or she is looking for is available in a different retail store, and not the store at which he or she is present. Therefore, in this sort of Cyberspace environment, it will be oftentimes desirable to embed a CPIR-enabling Applet in the home-page (or other conspicuous locations) of each retailer's WWW EC store so that, upon clicking the graphical icon thereof, an independent Java GUI to the UPC Request Retailer WWW site "@ the

retailer store" will be automatically produced so that all modes of searching are made available to the consumer against only those manufacturers registered (and possibly unregistered) with the UPN/URL Database Management Subsystem **9** which supply consumer products for sale within the particular retail store, as shown in **FIG. 402**. Notably, this Java GUI is similar to the Java GUI set forth in **FIG. 3**C, except that a "manufacturer filter" set by the retailer UPC product catalog is used to filter out the search results displayed on the Java GUI.

[0549] Referring to FIGS. **4**PI and **4P2**, it can be seen that the consumer within the EC-enabled store shown in FIGS. 4O1 and 4O2 has proceed to look at a particular product in the retail store (e.g. the "Ultralite Dagger Mountain Bike" being offered for sale for $285.00). At this point of presence within the EC-enabled retail store, the consumer might like to review the very best information published wherever on the WWW relating to this particular consumer product. Therefore, in this sort of Cyberspace environment, it will be desirable to embed a CPIR-enabling Applet within or near the image of this product in the retailer's WWW EC store so that, upon clicking the graphical icon thereof, a "UPC Request Cyber-Service URL Search" will be automatically carried out within the UPN/URL Database Management Subsystem **9**, and the search results thereof displayed in a Java GUI, as shown in **FIG. 4P2**. As shown, the Java GUI displays a menu-formatted list of categorized URLs that have been symbolically linked to the UPN of the consumer product on which the search inquiry was initiated. Typically, this menu of URLs, accessed from the UPN/URL Database Management Subsystem **9**, would have been updated as early as the night before during UPN/URL link updating/management operations carried out between (i) the UPN/URL catalog maintained in a client computer subsystem **13** within the backoffice of the manufacturer, and (ii) the Manufacturer/Product Registration Subsystem **31, 33**, using electronic data interchange processes based on any one of number of protocols (e.g. ftp, EDI, XML/ICE, etc.).

[0550] Referring to FIGS. **4Q1** through **4Q2**, the above-described method of CPI display is illustrated in the context of a consumer visiting an on-line BC-enabled auction site (e.g. at http://www.ebay.com), and considering whether or not to place a bid on a particular consumer product displayed within the auction listings thereof. In general, this environment is similar to the situation where a consumer finds him/herself searching for consumer product information at a WWW Search Directory or Engine, such as Yahoo, Excite, Alta Vista, Lycos, etc. In such an environment, it will be desirable for the consumer to search against all manufacturers within the entire UPN/URL Database Management Subsystem **11** before returning the search results to the consumer for display. Therefore, in this sort of Cyberspace environment, it will be oftentimes desirable to embed a CPIR-enabling Applet in the home-page of the WWW on-line auction site so that, upon clicking the graphical icon thereof, a n independent Java GUI to the UPC Request Central WWW site will be automatically produced so that all modes of searching are made available to the consumer against all manufacturers registered (and possibly unregistered) within the UPN/URL Database Management Subsystem **9**, as shown in **FIG. 4Q2**. Notably, this Java GUI is very similar to the Java GUI set forth in **FIG. 3C**.

[0551] Referring to **FIG. 4**RI and **4R2**, it can be seen that the consumer within the on-line auction site shown in FIGS. **4Q1** and **4Q2** has proceed to look at a particular item being auctioned off (e.g. the "Sony Mavica MVC-FD81" at a current bid of $420.50). At this point of presence within the on-line auction site, the consumer might very well like to review the very best information published wherever on the WWW relating to this particular consumer product. Therefore, in this sort of Cyberspace environment, it will also be desirable to embed a CPIR-enabling Applet within or near the title of the product being auctioned (or image thereof if available) so that, upon clicking the graphical icon thereof, a "UPC Request Cyber-Service" URL Search will be automatically carried out within the UPN/URL Database Management Subsystem **9**, and the search results thereof displayed in a CPID-enabling Java GUI, as shown in **FIG. 4R2**. As shown, this Java GUI displays a menu-formatted list of categorized URLs that have been symbolically linked to the UPN of the auctioned consumer product on which the search inquiry was initiated. Typically, this categorized menu of URLs, accessed from the UPN/URL Database Management Subsystem **9**, would have been updated as early as the night before during daily UPN/URL link updating/management operations carried out in the manner described hereinabove.

[0552] Referring to FIGS. **4S1** through **4S2**, the above-described method of CPI searching and display is illustrated in the context of a consumer visiting a typical WWW site (e.g. the Applicant's Intellectual Property Law Firm at http://www.tjpatlaw.com), whereupon an Internet advertisement is presented for a particular consumer product, solely for illustrative purposes. At this point of presence on the WWW, the consumer might very well like to review information published on the WWW relating to the advertised consumer product. Therefore, in this sort of Cyberspace environment, it will also be desirable to embed a CPIR-enabling Applet within, closely near, or immediately about the space of the advertisement so that, upon clicking the image associated thereof, a "UPC Request Cyber-Service" URL Search will be automatically carried out within the UPN/URL Database Management Subsystem **9**, and the search results thereof displayed in a CPID-enabling Java GUI, as shown in **FIG. 4S2**. As shown, this Java GUI displays a menu-formatted list of categorized URLs that have been symbolically linked to the UPN of the advertised consumer product on which the search inquiry was initiated. Typically, this categorized menu of URLs, accessed from the UPN/URL Database Management Subsystem **9**, would have been updated as early as the night before UPN/URL link updating/management operations carried out in the manner described hereinabove.

[0553] In situations where the advertisement itself embodies a Java-Applet, as in the case of most banner-type advertisements, it would be desirable to embed the CPIR-enabling Applet within the HTML-encoded document displayed within the new Java GUI generated when the Java-Applet is executed by the consumer upon his or her initial encounter of the advertisement. Upon the display of the menu-formatted list of categorized URLs within the CPID-enabling Java GUI, the consumer can easily access different Web-documents containing information related to the advertised consumer product by simply selecting the URL and linking to the information resource to which it points on the WWW. Notably, the displayed URL menu would include (i) one or more URLs pointing to EC-enabled stores and on-line

catalogs at which the advertised product can be purchased over the Internet, as well as (ii) one or more URLs pointing to "brick and mortar" type retail stores at which the advertised product can be purchased in the stream of commerce.

[0554] As shown in **FIG. 4S3**, the consumer having accessed the product-specific search results of **FIG. 4S2**, may then select, from the displayed URL Menu, a URL displayed in the "Buy On The Web" URL category thereof, thereby automatically linking to the EC-enabled store or product catalogue specified by the selected URL, as shown in **FIG. 4S2**, and thus enabling the purchase of the advertised product or service thereat. Preferably, the EC-enabled store or product catalog employs the "one-click purchase order" placement system and method taught in U.S Pat. No. 5,960,411 to Hartman, et al., and assigned to Amazon.com, Inc., which is incorporated herein by reference in its entirety. This would simplify ordering the product by the retailer having the consumer's credit card and shipping address information on file.

[0555] Thus, the CPI-based search and display method of the present invention gives rise to a new method of and system for purchasing consumer products over the Internet (e.g. WWW) comprising the steps of: embedding a UPN-encoded CPIR-enabling Applet within the HTML-code of a consumer product advertisement, wherein the CPIR-enabling Applet, when executed, automatically displays a categorized URL menu containing one or more URLs pointing to one or more EC-enabled stores or on-line catalogs on the WWW at which the consumer product identified by the encoded UPN can be purchased and delivered to a particular address in physical space.

[0556] Referring to FIGS. **4T1** through **4T2**, the above-described method of CPI searching and display is illustrated in the context of a consumer visiting a particular on-line electronic trading WWW site (e.g. http://www/etrade.com). At this site, the consumer is assumed to be reviewing the performance chart of a particular consumer product company displayed at this electronic trading WWW site, and is considering whether or not to buy, keep or sell securities (e.g. stock or bonds) in this consumer product company. At this point of presence on the WWW, the consumer decides that he or she would like to first ascertain specific information about the company's products by initiating a trademark/company name-directed CPI search according to the principles of the present invention. In accordance with the present invention, this would be achieved by the consumer identifying a client-side or server-side CPIR-enabling Java Applet embedded within the HTML code of the performance chart displayed at the on-line electronic trading WWW site. In the illustrated embodiment, the CPIR-enabling Applet is graphically indicated by a n associated graphical image (e.g. UPC Request™ Cyber-Service™ Trademark-Directed URL Search) and is encoded with the trademark an/or company name of a particular manufacturer/vendor associated with the display performance chart. Notably, the creation, distribution and embedding of such CPIR-enabling Applets must be carried out well in advance of the consumer arriving at the particular point of presence shown in **FIG. 4T1**. In accordance with the principles of the present invention, when the consumer performs a single mouse-clicking operation on the graphical image associated with the embedded CPIR-enabling Java Applet, the underlying CPIR-enabling Applet is executed and a trademark-directed URL search is automatically made against the UPN/URL Database Management Subsystem **9** hereof. Quickly thereafter, the results from the trademark/company name directed search are automatically displayed in a Java GUI on the browser of the requesting consumer's client machine, as shown in **FIG. 4T2**. As shown, the consumer is free to scroll through the displayed GUI, looking for URLs on particular consumer products of the manufacturer/vendor.

[0557] Preferably, in above application, each entry in the displayed Trademark Search Results screen shown in **FIG. 4T2** is itself a CPIR-enabling Java Servlet which, when clicked upon, automatically initiates a UPN-directed CPI search against a particular product of the manufacturer related to the displayed stock performance chart, as taught in great detail hereinabove. This novel technique will greatly simplify accessing and displaying accurate and up-to-date UPC/URL menus on the products offered by a particular company in which a consumer is considering buying, keeping or selling a particular number of financial securities. Also, while conducting such on-line CPI research, the consumer may also consider purchasing a particular consumer product at an EC-enabled store or product catalog, as illustrated in **FIG. 4S3**, supra.

Overview of Modes of Operation for IPI Finding and Serving Subsystem

[0558] In order to enter a primary mode of operation of the IPI Finding and Serving Subsystem, the consumer, retail sales clerk or retailer selects a particular mode activation button (e.g. **21A**, **21B**, **21C**, **21D**, **21E**, or **21F**) displayed in the control frame **21B** of the Java GUI browser program at the requesting client subsystem **13**. Upon making the selection, the Web browser at the client subsystem **13** automatically requests a particular HTML-encoded form (typically residing on the IPD Server(s) **11**). In general, each mode activation button **21A** through **21F** can be linked to a client-side or server-side Java Applet tag embedded within an HTML-encoded document, or directly to a predefined static-type HTML form corresponding to the selected mode of operation. In the case of Java Applets, upon selecting the mode selection button, a Java GUI is automatically produced and displayed within the information display frame **20C** of the Web browser of the requesting client subsystem. In the case of the directly-linked static-type HTML forms, a GUI in the form of HTML document is automatically produced and displayed within the information display frame **20C** of the Web browser of the requesting client subsystem. In either case, the HTML-encoded form corresponds to the selected mode and is linked to a Java method (or CGI script) related to the selected mode and possibly to other methods or forms required to carry out the database access and/or management process associated therewith. The requesting client subsystem then enters the information requested by the HTML form displayed within the information display frame **20C** of the Web browser's GUI interface. Information entry into the HTML form can be carried out using bar code symbol reading equipment, keyboard or keypad, speech dictation equipment (by Dragon Systems, Inc. of Newton, Mass.), and the like.

[0559] In general, the particular messages which will be displayed within the HTML forms during any particular mode of operation will depend upon several factors namely: whether the EPI Web-site is intended for access by bar code

driven kiosks (i.e. client subsystems **13**) as shown, for example, in FIGS. **3A2**, **3A3**, **3A4**, and **3A5** located within retail environments; or whether the IPI Web-site is intended for access by desktop, laptop and palmtop client computer systems **13** as shown, for example, in **FIG. 3A1** located at home, in the office or on the road.

[0560] For example, if the IPI Web-site supported by the IPI Finding and Serving Subsystem hereof is intended for access by bar code driven kiosks, then the HTML documents related to the IPI Website will be particularly adapted to facilitate the use of bar code symbol reader at the client subsystem. This way UPNs (e.g. UPC or EAN symbols) can be easily entered into the subsystem without manual key-entry operations. In contrast, if the IPI Web-site supported by the IPI Finding and Serving Subsystem hereof is intended for access by client subsystems not having bar code symbol readers (e.g. Web enabled computer systems at home, in the office or on the road), then the HTML documents related to the IPI Website will be particularly adapted to facilitate the use of data-entry display screens at the client subsystem. This way, UPNs (e.g. UPC or EAN symbols) can be easily entered into the subsystem using bar code symbol scanners avoiding manual key-entry operations. In the illustrative embodiment, bar code-code driven and manual data-entry IPI Websites are served from a "framed" Java GUI, in which the control strip **20B** has six (5) Check Boxes **21A** through **21F** described above to enable the consumer, retail sales/service personnel as well as manufacturers to select the particular mode of operation that suits his or her consumer product information needs at any particular instance in time.

[0561] It understood that the use of Java Applets (including Servlets) will be most beneficial in constructing Java-based IPI Central and retail WWW sites, as indicated above, and in most instances will be preferable over static HTML documents and CGIs linking the IPD (http) server **11** to the backend RBDMS servers **9** of the system. However, for purposes of illustration only, the six primary modes of operation of the system will be described below using a CGI implementation, illustrated in **FIG. 2B2**. However, it is understood that implementations using CPIR-enabling Servlets as shown in **FIG. 2B1** can be used to replace such CGI constructions. Also, implementations using CPIR-enabling Applets as shown in FIGS. **2B3** and **2B4** can be used to enable access to the UPN/URL Database Management Subsystem **9** and its supporting RDBMS servers.

Manufacturer/Product Registration Mode of Operation

[0562] Referring to **FIG. 5A**, the high level structure is shown for a communication protocol that can be used among a client subsystem C_a, an IPD Server Sb, and an IPI Server Sc of the IPI finding and serving subsystem hereof when it is induced into the Manufacturer/Product Registration Mode of operation from the point of view of the depicted client subsystem. **FIG. 6A** provides a high level flow chart illustrating the steps involved in carrying out this communication protocol when the IPI Finding and Serving Subsystem is in its Manufacturer/Product Registration Mode of operation.

[0563] As indicated at Block A in **FIG. 6A**, when selected from the user interface of an IPI Website, the first Check Box type button **21A** automatically activates the Manufacturer/Product Registration Mode of the IPI Finding and Serving Subsystem by sending an HTTP request to the IPD Server(s) **11"** based on a URL hot-linked to the selected Check Box. As indicated at Block B in **FIG. 6A**, this causes a HTML-encoded document residing on the IPD Server **11"** shown in **FIG. 2B2**, to requesting client subsystem **13** or display on the information display frame **20C** thereof. The HTML document of the illustrative embodiment displays several types of information relevant to the Manufacturer/Product Registration Mode, namely: eligibility requirements (i.e. qualifications) for a manufacturer to register with the IPI Finding and Serving Subsystem; optional ways of registering consumer products and product-related information with the Manufacturer/Product Registration Subsystem hereof **33**; ways of acquiring computer software necessary for managing consumer product-related information (e.g. UPNs, URLs, trademarks and product descriptors) on a particular computing platform using EDI (or XML/EDI) techniques supported by the Manufacturer/Product Registration Subsystem **33**; etc; and one or more Check Boxes embodying links (i.e. anchors) to HTML documents, CGI scripts and the like designed to facilitate this mode of operation. Notably, at least one of these HTML documents will be located on the Web Document Server **30** of the Manufacturer/Product Registration Subsystem **33**, providing manufacturers (and/or their designated information-managers and agents) with a point of entry into the manufacturer/product registration process hereof. As indicated at Block C in **FIG. 6A**, the manufacturer and or its agent follow the instructions displayed on the HTML document, linking to the Web Document Server **30** of the Manufacturer/Product Registration Subsystem **33** and filling out the various HTML forms transmitted to the requesting client subsystem, downloading Web-based EDI (or XML/EDI) software for UPN/URL management; and the like. While carrying out registration of manufacturers with the subsystem is relatively straightforward, there are a number of different ways of carrying out the Product Registration Mode of the subsystem. These alternative techniques will be described below.

[0564] The first method illustrated in FIGS. **2-1** and **2-2** involves by carrying out FTP between a client subsystem of the registering manufacturer (or its agent) Mi and IPD Server **11"** in order to update the IPI Registrant Database associated therewith. This can be carried out by the manufacturer's officer or agent surfing to the IPI Website, selecting the "Product Registration Mode" from the control strip, and then following the instructions displayed on the various screens of the Website in this mode. When using the first method, product UPCs, URLs and other information elements can be formatted within suitable Product Registration Forms and transmitted by FTP from the client subsystem or Database Server of a registering manufacturer to the IPD Server **11"** so that the IPI Registrant Database thereof can be updated accordingly. The first method will be desirable typically when registering a few consumer-products.

[0565] The second method illustrated in **FIG. 2A**, involves first carrying out EDI between a client subsystem of the registering manufacturer (or its agent) and the UPN/URL Database Subsystem **9**, and then carrying out FTP or SMTP between the client subsystem and IPD Server **11"** in order to update the IPI Registrant Database maintained therein. The second method will be desirable when a manufacturer needs or desires to register a large number of

consumer-products. The details of these information transmission methods will be described below.

[0566] When using the second method, conventional EDI protocols or more modern protocols (e.g. XML/ICE) can be used to transmit product UPCs, URLs and other information elements from client subsystems or database servers of manufactures to the UPN/URL Database Subsystem **9** of the present invention. FTP can be used to transmit UPCs and URLs from the UPN/URL Database Subsystem to each IPD Server in the system so that the IPI Registrant Database thereof can be updated accordingly. Once registered with the system using either of these methods in the Product Registration Mode, such consumer-products can be easily found on the Internet by anyone wishing to use the product finding techniques of the present invention.

[0567] The third method involves by carrying out electronic data and document interchange over the Internet between the WebDox Remote™ Computer System **13** and the WebDox™ Server **30** of the system of the present invention, and communication between the WebDox™ Server **30** and the WebDox™ Admin computer system **31** of the system hereof. The various steps involved in this embodiment of the consumer product registration process will be described in detail below.

[0568] When the manufacturer selects the "Product Registration Mode" of the system, a Manufacturer Registration Form is automatically downloaded from the WebDox™ Server **30** to the Manufacturer's client computer system **13** (i.e. the WebDox Remote™ Computer System). At the end of the downloading process, a Manufacturer Registration Form is presented (i.e. displayed) and the manufacturer then enters some requested identification information (e.g. Manufacturer's Company Name, Address, Name of CEO and President, phone number, 6-digit Manufacturer Identification Number assigned by ,the UCC, etc.) and presses the "Send" button on the Manufacturer Registration Form. The form is then transmitted immediately via the Internet and received by the WebDow™ Server **30**. At the WebDow™ Server **30**, an automated process takes the information in the Manufacturer Registration Form and registers the Manufacturer with the system.

[0569] Upon registering the manufacturer with the system, the manufacturer is asked to select which version of "customized" WebDox Remote™ software (i.e. the UPN/URL Registration Application) the manufacturer would like downloaded to its client computer system **13** (e.g. WebDox Remote with UPN/URL Database and CGI scripts for MacOS WebServer, WebDox Remote with UPN/URL Database and CGI scripts for UNIX Web Server, or WebDox Remote with UPN/URL Database and CGI scripts for NT Web Server). Once the manufacturer makes its selection, the customized WebDox Remote software is automatically downloaded to the manufacturer's client computer system **13**. This downloaded software includes a computer program that automatically generates (on the. manufacturer's) client subsystem, a relational database management system (RDBMS) which allows the manufacturer (or its agents) to easily construct and maintain a UPN/URL database (akin to that specified in **FIG. 4A1**) but restricted to containing information relating only to the manufacturer's products. Thus, when the manufacturer attempts to enter a UPC number into the manufacturer's UPN/URL database that does not contain the 6-digit Manufacturer Identification Number assigned to the manufacturer by the UCC, the RDBMS automatically blocks all such information entries. Consequently, the UPN/URL database can only maintain information pertaining to the registered manufacturer's products and information relating thereto on the Internet. As the manufacturer adds or removes products from its retail or wholesale line, the database administrator simply adds or removes the UPC and URL information relating thereto from the RDBMS. As will be described in greater detail hereinafter, such database changes are periodically transmitted to the WebDow™ Server **30** so that the IPI Registrant Database (i.e. master UPN/URL database) of the system (maintained on the IPD Servers thereof) can be updated in a timely manner.

[0570] Preferably, the limited or restricted version of the UPN/URL database maintained by each registered manufacturer on its client subsystem **13** is connected to the manufacturer's Internet Server **12'** (or **12B**) by a CGI script or Java method, as shown in FIGS. **2-1** and **2-2**. In this way, the manufacturer's limited version of the UPN/URL database can be made accessible to consumers world-wide from the manufacturer's Website which, in the illustrative embodiment, is assumed to be hosted on an Internet information server **12'** or **12B** that is similar to an IPI Server **12** described in detail hereinabove. In order to simply the process of serving of the manufacturer's limited-version of the UPN/URL database on the WWW, it is preferred that the CGI script **40**, input forms, output forms, and methods for searching and the displaying the results from the limited-version UPN/URL database are predesigned for use with manufacturer's Internet Server **12'** (taking into consideration its operating system and the like). This way, prior to registration the manufacturer need only make a selection of the type of customized WebDox Remote software it needs for its computing and Internet serving platform(s). Then, during software download, the WebDox Server **30** simply transmits the suitable version of the customized WebDox Remote software to the manufacturer so that it can create, maintain and serve (on the WWW) its limited version of the UPN/URL database in a "turn-key" manner.

[0571] In the illustrative embodiment of the present invention, the homepage of each registered manufacturer's Website will display a visually conspicuous radio button labeled "UPC Request™ Product Finder" or the like. Moreover, whenever a consumer attempts to search the manufacturer's limited-version UPN/URL database for products not registerable to the manufacturer (i.e. using UPC numbers not containing the manufacturer's 6-digit UCC Manufacturer Identification Number), the limited-version of the UPN/URL database will automatically display an HTML-encoded message from the manufacturer's Website, urging the consumer to surf to the IPI Registrant Database of the system (maintained on the network of IPD servers **11**). Preferably, such HTML-encoded messages will have a hot-linked URL (i.e. anchor) to Website(s) providing consumer access to the "master" UPN/URL database.

[0572] The WebDox Remote™ computer system **13** available to each registered manufacturer has both online and offline modes of operation. In the offline mode, the manufacturer responds to a UPN/URL Registration Request from the WebDow™ Server in the following manner. First, the WebDox Remote™ software analyzes the limited-version of

the UPN/URL database that it has been currently created and maintained by the manufacturer or its designee. Thereafter, the WebDox software automatically creates a UPN/URL Registration Response document which contains a set of currently active URLs specifying the address location of Web-based information resources associated with each UPC-encoded product of the manufacturer. Then, WebDox Remote™ program establishes an Internet connection with the WebDow™ Server, through a "Get/Send Mail" option. This delivers the UPN/URL Registration Response (document) to the WebDox™ Server **30** and retrieves any documents which are waiting thereat for the manufacturer. These new documents are listed by WebDox Remote™ program and presented in the InBasket of the manufacturer's WebDox Remote™ computer system **13**.

[0573] In the online mode, WebDox Remote™ (under the control of the Form Application) can also send UPN/URL Registration Request documents immediately. For very sensitive applications (i.e. Just-in-Time), this ensures that the UPN/URL Registration Response document is received at the WebDox™ Server **30** the moment that the manufacturer completes the document.

[0574] In general, the WebDow™ Server **30** provides a high-volume document processing and mailboxing environment between the WebDox Server and the WebDox Remote™ system of each registered manufacturer. WebDox™ Server **30** performs: permanent storage and tracking of all UPN/URL Registration Request documents sent and UPN/URL Registration Response documents received; automatic reconciliation of acknowledgments from WebDox Remote™ program; automatic creation of user-friendly receipt messages to the manufacturer; "mailboxing" of outbound UPN/URL Registration documents for retrieval by manufacturer; and automatic manufacturer and profile creation based on forms received from manufacturers. The WebDox™ Server **30** consists of online components that run as extensions to Microsoft's Internet Information Server (IIS) using the ISAPI interface. This provides higher performance and lower hardware requirements than a conventional CGI Web Interface. Processing intensive tasks are performed asynchronously from the Web server. An integrated queuing and dispatching system manages the processing of documents and interaction with the corresponding application. For large volume situations, the WebDow™ Server components can be deployed on different machines, the WebDow™ Server components (ISAPI extensions) on one machine, the processing components and database on another machine.

[0575] Data for UPN/URL Registration Request documents to be sent to manufacturers is extracted from the IPI Registrant Database using an interface or utility program. The document data (e.g. information fields associated with UPN/URL registration) can then be accepted by WebDox in a direct manner after formatting. The UPN/URL Registration Request document should be formatted to a file structure created during the design of the UPN/URL Registration Application. The WebDox™ Server **30** then converts the application data into a UPN/URL Registration request document (i.e. data package). The data package for each manufacturer is then stored (as a message) in an assigned Mailbox of the WebDow™ Server **30**. These messages are then available to be retrieved by the registered manufacturers using WebDox Remote's™"Get/Send Mail" feature.

[0576] As discussed above, the WebDox Remote™ program transmits messages (e.g. UPN/URL Registration Response documents) to the WebDow™ Server **30**, where, after passing security checks, they are placed in the WebDox Mailbox system. Incoming (document) messages are received from the Mailbox, processed, and converted into data files for direct transfer to the database management system handling the IPI Registrant Database.

[0577] For each document received, the WebDow™ Server **30** will return a message to the manufacturer confirming receipt of the document. WebDox Remote™ system also returns delivery confirmations to the WebDox Server. These messages are used by the WebDow™ Server to track the status of messages. WebDox™ Server **30** maintains Mailbox Files for all inbound and outbound messages. The status of messages is updated on an ongoing basis as acknowledgement messages are received, allowing timely and precise audits.

[0578] WebDox Admin™ Computer system **31** provides an easy-to-use tool to manage the community of manufacturers, review the status of documents, and configure the WebDow™ Server **30**, including: ad hoc maintenance of manufacturer information; online display of the Mailbox permitting inquiry into document status or document activity for particular manufacturers, and the ability to reset document status; creation and maintenance of UPN/URL Registration Profiles; preparation of "releases" of new and updated UPN/URL Registration Applications; Distribution of new and updated UPN/URL Registration Applications; and automatic inventory and tracking of UPN/URL Registration Applications distributed to manufacturers.

[0579] In the preferred embodiment, UPN/URL Registration Application design and development is carried out on a Windows 95 or NT workstation. The UPN/URL Registration Application is developed, tested, and then fully implemented for production with manufacturers. New or updated UPN/URL Registration Applications are registered with the WebDox Admin™ computer system **31** and are then distributed to the manufacturers as described herein above.

[0580] In the preferred embodiment, UPN/URL Registration Applications are developed using Microsoft Visual Basic™ and related software tools. These products provide rapid design and creation of the screen-based forms that the manufacturer uses. In addition, the "intelligence" behind the form, in the UPN/URL Registration Application, can be very powerful, making the manufacturer's work easier while ensuring that the user and Server application receive high quality data.

[0581] The WebDox Admin™ system handles the distribution of UPN/URL Registration Applications to manufacturers. New UPN/URL Registration Applications can be sent to some or all of the existing manufacturers assigned UPC Manufacturer Identification Numbers. Updates to UPN/URL Registration Applications can be sent to manufacturers who are currently using that UPN/URL Application. The actual update is distributed by sending a small notification message to each manufacturer, which then results in the remote site downloading the new forms from the WebDow™ Server **30**, as hereinbefore described above.

[0582] Notably, the WebDoc™ Solution has been described above provides one way and means of implement-

ing a method of electronic data and document interchange between client machines of manufacturers and the IPI Registrant Database (i.e. master UPN/URL database in subsystem **9**) of the system of the present invention. It is understood, however, that many different types of electronic data interchange solutions (e.g. XML or XML/EDI) can be used to practice the system and method of UPN/URL database management in a n efficient and timely manner so that consumers will always be provided with up-to-date URL links on the Internet. For example, the new CenterStage 4 Application Suite from On Display, Inc. of San Ramon Calif., can be used to enable XML-based electronic data interchange (i.e. transfer) between the client computer subsystems **13** operated within the backoffices of manufacturers, and the IPI Registrant Database (i.e. master UPN/URL database) of the system hereof operated in the backoffice of the system administrator. Manufacturers (i.e. vendors) can format their data transactions in any of the many new languages of electronic-business (e.g. cXML, RosettaNet, CBL, BizTalk, OBI, ICE proprietary formats, or standard EDI formats such as ANSI X12), and the CenterStage 4 platform will automatically convert their transactions into the chosen formats of the system administrator responsible for managing the master UPN/URL database.

[0583] For further details on the use of electronic data interchange technologies in order to realize this functionality of the system of the present invention, reference can be made to the following technical publications: "XML/EDI: Cyber Assisted Business in Practice" (1999) by Dick Raman (ISBN: 90-8050233-2-1); The A to Z of EDI and Its Role in E-Commerce" Second Edition, 1998) by Nahid Jilovec, published by Duke Communications, Inc., Loveland, Colo.; "Electronic Commerce With EDI: A Guide For Decision Makers" (1998), by Robert L. Sullivan, published by Twain, Inc. North Andover, Mass.; and "Wild's WWW: Technical Foundations of the World Wide Web" (1999) by Erik Wilde, published by Springer-Verlag, Berlin, Heidelberg; each said publication incorporated herein by reference as of set forth herein.

[0584] In **FIG. 2C**, there is disclosed a novel distributed method of collecting, managing and transmitting UPN/URL menus for consumer products. Notably, this distributed system and method will be useful in large corporate environments, where departmentalization is the general rule. As shown, instead of each manufacturer having a single EDI-enabled workstation (equipped with EDI or EDI/XML software) **13** for carrying out UPN/URL management operations, a group of EDI-enabled client computers **13** are connected to a local or wide area network **200** via a network-centric Web (http) server **133** using a network router **201** to interface with the infrastructure of the Internet, as well as the other local or wide area network **200** as shown in **FIG. 2C**. Preferably, each client computer **13** on the LAN or WAN is equipped with UPN/URL management software for managing the consumer product information collected in the UPN/URL Database for a particular manufacturer, as shown in FIGS. **4A1** through **4B**.

[0585] In one arrangement, each manufacturer-operated client machine **13** would be assigned the task of managing the UPN/URLs associated with a particular department of the manufacturer (e.g. engineering department, sales department, service/support department, marketing department, advertising department, etc.). The UPN/URLs menus and other CPI related information collected by each department is maintained within a local UPN/URL Database **202** on the department's client machine **13**, and is periodically transmitted to a Manufacturer's UPN/URL Database **203** hosted on the network Internet server **133** In addition to providing the client machine behind the corporate firewall with http, e-mail and ftp services, the network Internet server **133** is also equipped with an EDI (e.g. EDI or XML/ICE) software solution which enables periodic uploading of the manufacturer's UPN/URL Database **203** to the Central UPN/URL Database Management Subsystem **9**, shown in **FIG. 2C**

[0586] Another arrangement, each manufacturer-operated client machine **13** would be assigned the task of managing the UPN/URLs associated with a particular department of the manufacturer (e.g. engineering department, sales department, service/support department, marketing department, advertising department, etc.). The UPN/URLs menus and other CPI related information collected by each department is maintained within a local UPN/URL Database **202** on the department's client machine **13**, and is periodically transmitted directly to the Central UPN/URL Database Management Subsystem **9**, shown in **FIG. 2C**. In such an alternative embodiment of the present invention, the network Internet server **133** would provide each client machine behind the corporate firewall with http, e-mail and ftp services in a conventional manner, but not maintain a central manufacturer's UPN/URL database **202**.

[0587] The primary advantage of the above described techniques for distributed UPN/URL management hereof is that such techniques provide manufacturers with a revolutionary way of am and means for enlisting the different departments within the organization, having different business perspectives, goal and resources, to create "up-to-date" links between UPN's on their consumer products and the diverse types of consumer related information resources published on the Internet, all in concerted effort to achieve the sales, marketing and support programs of the company in a unified manner. Using the system and method of the present invention, symbolic links between the manufacturer' products and published information resources on the Internet (e.g. WWW) can be impressed upon the minds of consumers as they seek access to such current information at home, in the office, in physical and electronic stores, as well as on the road.

Manufacturer Website Search Mode of Operation

[0588] Referring to **FIG. 5B**, the high level structure is shown for a communication protocol that can be used among a client subsystem Ca, an IPD Server Sb, and an IPI Server Sc of the IPI finding and serving subsystem hereof when it is induced into the Manufacturer Website Search Mode of operation from the point of view of the depicted client subsystem. **FIG. 6B** provides a high level flow chart illustrating the steps involved in carrying out this communication protocol when the IPI Finding and Serving Subsystem is in its Manufacturer Website Search Mode of operation.

[0589] As indicated at Block A in **FIG. 6B**, when selected from the user-interface of a bar-code driven IPI Website, the second Check Box type button **21B** automatically activates the Manufacturer Website Search Mode of the IPI Finding and Serving Subsystem by sending an HTTP request to the IPD Server(s) **11**" based on a URL hot-linked to the selected Check Box.

[0590] As indicated at Block B in FIG. 6B, this causes a particular type of HTML-encoded document (i.e. called an "HTML form" or "Web form document") residing on the IPD Server(s) 11" to be sent to the Web browser of the requesting client subsystem 13 and displayed on the information display frame 20C thereof (requesting this mode of service). As in the Manufacturer Website Search Mode described above, the HTML form sent in the Manufacturer Website Search Mode may also use any HTML format commands, such as headers, paragraphs, and lists, but must include three unique items, namely: the METHOD by which the user input is to be sent; the ACTION, which specifies a URL to which the user input is to be sent, presumably, the IPD Server 11" that will act upon the request appropriately; and a SUBMIT button, to send the completed form over the Internet via HTTP. In the illustrative embodiment, user input (i.e. a UPC or EAN number associated with a particular consumer product) is obtained by an Input Box, which allows the user (i.e. retail sales clerk or consumer) to type in or scan in a UPC or EAN number assigned to a consumer product on which product related information is sought.

[0591] As indicated at Block C in FIG. 6B, the consumer or retail clerk scans the bar coded consumer product, or enters the UPC or EAN number thereon into the Input Box of the HTML form, and selects the SUBMIT button thereon. In response thereto, the Web browser on the client subsystem 13 sends a GET request to the IPD server 11B" shown in FIG. 2B2. When selecting the SUBMIT button on the HTML form, the Web browser executes the METHOD associated with the HTML form and sends the stored UPC (or EAN) value to the URL specified by ACTION associated with the HTML form (i.e. the Web browser performs the action specified in the ACTION). The ACTION of the HTML form specifies the URL of the CGI script within the http server 11" that will process the request from the HTML form. This amounts to the Web browser constructing a GET request for that URL, with the arguments (the query string) being attached to the end of the URL. The arguments of the HTML form are specified by the INPUT items of the HTML form (i.e. the UPC or EAN number on the consumer product on which information is sought).

[0592] In general, the HTTP and HTML protocols provide three ways to pass the input (e.g. UPC or EAN number) from the users to CGI scripts on the IPD Server 11 (i.e. HTTP Server). All three CGI scripts accomplish the same thing: they allow the Web browser to pass information to the IPD Server 11".

[0593] As indicated at Block D in FIG. 6C, the HTTP (httpd) program on IPD server 11" passing the arguments (the UPC or EAN numbers in the query string) to the CGI script thereon and the CGI script translates the query string into a proper query for use in searching the RDBMS 9 shown in FIG. 2B2.

[0594] As indicated at Block E in FIG. 6C, the translated query is used to search the RDBMS 9 in order to find the set of URLs pointing to HTML documents (i.e. Web Pages) published on the Internet and containing information relating to the consumer product having the input UPC or EAN number. The result returned from the RDBMS 9 is an ASCII record specifying the set of URLs pointing to HTML documents published on the Internet and containing information relating to the consumer product having the input UPC or EAN number entered into the HTML form. In order for the Web browser of the requesting client subsystem to display the results of the database search using the UPC or EAN input, the ASCII record must be converted into a HTML document (i.e. output HTML form). As indicated at Block F in FIG. 6B, the IPD Server 11" creates the elements of an output HTML form (Web output form), inserts the result from the RDBMS 11" into the output form, and sets the Content-type to be text/html. The CGI script may translate, filter, augment and reformat the result from the database search in any way so long as the result is an HTML document or some format that the Web browser of the client subsystem can display.

[0595] As indicated at Block G, the menu of URLs retrieved from the database search is displayed in the Web output form. At Block H, the consumer or retail sales clerk can link to a desired consumer product related information resource (HTML document) by selecting from the information menu, the URL anchored to the information resource in the displayed information menu.

UPN-Directed Information Access Mode of Operation

[0596] Referring to FIG. 5C, the high level structure is shown for a communication protocol that can be used among a client subsystem Ca, an IPD Server Sb, and an IPI Server Sc of the IPI finding and serving subsystem hereof when it is induced into the UPN-Directed Information Access Mode of operation from the point of view of the depicted client subsystem. FIG. 6C provides a high level flow chart illustrating the steps involved in carrying out this communication protocol when the IPI Finding and Serving Subsystem is in its UPN-Directed Information Access Mode of operation.

[0597] As indicated at Block A in FIG. 6C, when selected from the user-interface of an IPI Website, the third Check Box type button 21C automatically activates the UPN-Directed Information Access Mode of the IPI Finding and Serving Subsystem by sending an HTTP request to the IPD Server(s) 11 based on a URL hot-linked to the selected Check Box.

[0598] As indicated at Block B in FIG. 6C, this causes a particular type of HTML-encoded document (i.e. called an "HTML form" or "Web form document") residing on the IPD Server(s) 11" to be sent to the Web browser of the requesting client subsystem 13 and displayed on the information display frame 20C thereof (requesting this mode of service). As in the Manufacturer Website Search Mode described above, the HTML form sent in the UPN-Directed Information Access Mode may also use any HTML format commands, such as headers, paragraphs, and lists, but must include three unique items, namely: the METHOD by which the user input is to be sent; the ACTION, which specifies a URL to which the user input is to be sent, presumably, IPD Server 11' that will act upon the request appropriately; and a SUBMIT button, to send the completed form over the Internet via HTTP. In the illustrative embodiment, user input (i.e. a UPC or EAN number associated with a particular consumer product) is obtained by an Input Box, which allows the user (i.e. retail sales clerk or consumer) to type in or scan in a UPC or EAN number assigned to a consumer product on which product related information is sought.

[0599] As indicated at Block C in FIG. 6C, the consumer or retail clerk scans the bar coded consumer product, or

enters the UPC or EAN number thereon into the Input Box of the HTML form, and selects the SUBMIT button thereon. In response thereto, the Web browser on the client subsystem **13** sends a GET request to the IPD Server **11"** shown in **FIG. 2B2**. When selecting the SUBMIT button on the HTML form, the Web browser executes the METHOD associated with the HTML form and sends the stored UPC (or EAN) value to the URL specified by ACTION associated with the HTML form (i.e. the Web browser performs the action specified in the ACTION). The ACTION of the HTML form specifies the URL of the IPD SERVER **11** that will process the request from the HTML form. This amounts to the Web browser constructing a GET request for that URL, with the arguments (the query string) being attached to the end of the URL. The arguments of the HTML form are specified by the INPUT items of the HTML form (i.e. the UPC or EAN number on the consumer product on which information is sought).

[0600] As indicated at Block D, the IPD server **11"** passes the arguments (the UPC or EAN numbers in the query string) to a CGI script running therewithin which translates the Web query string into a proper query to the RDBMS **9** shown in **FIG. 2B1**.

[0601] As indicated at Block E, the translated query is used to search the RDBMS **11A** and find the set of URLs (i) linked to the registered consumer product (by the manufacturer or agent thereof) assigned the UPC or EAN number entered into the Input Box of the HTML form, and (ii) pointing to HTML documents on the WWW containing particular types of consumer product related information. The result returned from the RDBMS **9** is an ASCII record specifying the set of URLs satisfying the above criteria. In order for the Web browser of the requesting client subsystem to display the results of the database search during this mode, the ASCII record must be converted into a HTML document (i.e. Web output form).

[0602] As indicated at Block F, a CGI script within IPD server **11"** creates the elements of an HTML document (Web output form), inserts the result from the RDBMS **9** into the Web output form, and sets the Content-type of this HTML document to text/html. In the illustrative embodiment, when the Web output form is displayed by the requesting client subsystem, a set of URLs categorized by particular product information types is displayed on the information display frame **20C**. Notably, this set of URLs points to particular types of consumer product related information registered within the RDBMS **9** of the system.

[0603] As indicated at Block G, the consumer or retail sales clerk can then access and display any HTML document (Web page) located at a particular URL within the information menu by selecting the same using a touch screen, mouse, or other input selection device.

Trademark-Directed Search Mode of Operation

[0604] Referring to **FIG. 5D**, the high level structure is shown for a communication protocol that can be used among a client subsystem Ca, an IPD Server Sb, and an IPI Server Sc of the IPI finding and serving subsystem hereof when it is induced into the Trademark-Directed Search Mode of operation from the point of view of the depicted client subsystem. FIGS. **6D1** through **6D23**, taken together, provides a high level flow chart illustrating the steps involved in carrying out this communication protocol when the IPI Finding and Serving Subsystem is in its Trademark-Directed Search Mode of operation.

[0605] As indicated at Block A in **FIG. 6D1**, when selected from the user-interface of an IPI Website, the fourth Check Box type button **21D** automatically activates the Trademark-Directed Search Mode of the IPI Finding and Serving Subsystem by sending an HTTP request to the IPD Server(s) **11"** based on a URL hot-linked to the selected Check Box.

[0606] As indicated at Block B in **FIG. 6D1**, this causes a particular type of HTML-encoded document (i.e. called an "HTML form" or "Web input form document") residing on the IPD Server(s) **11"** to be sent to the Web browser of the requesting client subsystem **13** and displayed on the information display frame **21C** thereof (requesting this mode of service). As in the UPN-Directed Information Access Mode described above, the HTML form sent in the Trademark-Directed Search Mode may also use any HTML format commands, such as headers, paragraphs, and lists, but must include three unique items, namely: the METHOD by which the user input is to be sent; the ACTION, which specifies a URL to which the user input is to be sent, (e.g. a CGI script running within http server **11"** that will act upon the request appropriately); and a SUBMIT button, to send the completed form over the Internet via HTTP. In the illustrative embodiment, user input (i.e. the trademark or tradename used with a particular consumer product on which information is sought) is obtained by an Input Box, which allows the user (i.e. retail sales clerk or consumer) to type in the trademark or tradename believed or otherwise known to be used in connection with a particular consumer product on which information is sought.

[0607] As indicated at Block C in **FIG. 6D1**, the consumer or retail clerk enters the trademark or tradename into the Input Box of the HTML form, and selects the SUBMIT button thereon. In response thereto, the Web browser on the client subsystem **13** sends a GET request to the IPD server **11"** shown in **FIG. 2B2**. When selecting the SUBMIT button on the HTML form, the Web browser executes the METHOD associated with the HTML form and sends the stored trademark value to the URL specified by ACTION associated with the HTML form (i.e. the Web browser performs the action specified in the ACTION). The ACTION of the HTML form specifies the URL of the CGI script running within the IPD server **11"** that will process the request from the HTML form. This amounts to the Web browser constructing a GET request for that URL, with the arguments (the query string) being attached to the end of the URL. The arguments of the HTML form are specified by the INPUT items of the HTML form (i.e. the trademark or tradename used in connection with the consumer product on which information is sought).

[0608] As indicated at Block D in **FIG. 6D1**, the IPD server **11"** passes the arguments (the trademark or tradename in the query string) to a CGI script running therewithin and the CGI script translates the Web query string into a proper query to the RDBMS **9**, as shown in **FIG. 2B2**.

[0609] As indicated at Block E in **FIG. 6D1**, the translated query is used to search the RDBMS **9** in order to find all registered consumer products having product descriptions (PD) registered within the RDBMS **9** that are related to the

trademark or tradename entered into the Input Box of the HTML form. The result returned from the RDBMS **9** is an ASCII record specifying each triplet data set (Product Description, UPN and Manufacturer) which satisfies the above trademark search criteria (to a particular degree) entered into the Input Box of the HTML form. In order for the Web browser of the requesting client subsystem to display the results of the database search during this mode, the ASCII record must be converted into another HTML form for use in refining the consumer product information display.

[0610] At Block F in **FIG. 6D2**, a CGI script within IPD server **11"** creates the elements of another HTML document (Web auxiliary input form), inserts the preliminary search result from the RDBMS **9** into the Web auxiliary input form, and sets the Content-type of this HTML document to text/html. In the illustrative embodiment, the Web auxiliary-input form has an ACTION which specifies the URL of a CGI script within the IPD server **11"** that will act upon the request appropriately as if the system were in the UPN-Directed Information Access Mode. The Web auxiliary input form includes an Input Box listing all triplet data sets (i.e. Product Description, Manufacturers and UPN number) satisfying the input trademark search criteria entered in the primary Web input document, described hereinabove. The qualifying triplets listed in the Input Box are provided with a Radio-Button to allow the consumer or retail sales clerk to select one of the triplets from the list thereof for use in a subsequent refined search of the RDBMS **9**. The Web auxiliary-input form also has a SUBMIT button for sending the HTML form back to the IPD server **11"** for processing.

[0611] As indicated at Block G in **FIG. 6D2**, when the consumer or retails sales clerk makes a selection with the Radio-Button and then selects the SUBMIT button, the Web browser on the client subsystem **13** sends a request to the HTTP program on the IPD server **11"** to get the completed HTML form.

[0612] As indicated at Block H in **FIG. 6D2**, the HTTP (httpd) program on IPD server **11"** passes the arguments (the trademark or tradename in the query string) to a CGI script running therewithin and the CGI script translates the Web query string into a proper query for use in searching RDBMS **9**, as shown in **FIG. 2B2**.

[0613] At Block I in **FIG. 6D2**, the query is used to search the RDBMS **9** in order to find the set of URLs (i) related to the registered consumer product (by the manufacturer or agent thereof) assigned the UPN, (Product Description and Manufacturer) entered into the Input Box of the HTML (auxiliary) form, and (ii) pointing to HTML (or FTP) documents on the WWW containing particular types of consumer product related information. The result returned from the RDBMS **9** is an ASCII record specifying the set of URLs satisfying the above criteria. In order for the Web browser of the requesting client subsystem to display the results of the database search during this mode, the ASCII record must be converted into an HTML document (i.e. Web output form).

[0614] At Block J in **FIG. 6D2**, the IPD Server **11"** creates the elements of an HTML document (Web output form), inserts the result from the RDBMS **9** into the Web output form, sets the Content-type of this HTML document to text/html, and sends the HTML form to the requesting client subsystem.

[0615] At Block K in **FIG. 6D3**, the set of URLs categorized by particular product information types is displayed within the output HTML form on the information display frame **20C**. Notably, this set of URLs points to particular types of consumer product related information registered within the RDBMS **9** of the system by the manufacturer of the product or its agent(s) thereof using the UPN/URL management tools accessible during the Manufacturer/Product Registration Mode hereof.

[0616] As indicated at Block L in **FIG. 6D3**, the consumer or retail sales clerk can access and display any HTML document (Web page) located at a particular URL within the displayed information menu by selecting the same using a touch screen, mouse, or other input selection device available at the requesting client subsystem **13**.

Product-Description Directed Mode of Operation

[0617] Referring to **FIG. 5E**, the high level structure is shown for a communication protocol that can be used among a client subsystem Ca, a n IPD Server Sb, and an IPI Server Sc of the IPI Finding and Serving Subsystem hereof when it is induced into the Product-Description Directed Mode of operation from the point of view of the depicted client subsystem. FIGS. **6E1** through **6E3** provides a high level flow chart illustrating the steps involved in carrying out this communication protocol when the IPI Finding and Serving Subsystem is in its Product-Description Directed Mode of operation.

[0618] As indicated at Block A in **FIG. 6E1**, when selected from the user-interface of an IPI Website, the fifth Check Box type button **21E** automatically activates the Product-Description Directed Search Mode of the IPI Finding and Serving Subsystem by sending an HTTP request to the IPD Server(s) **11"** based on a URL hot-linked to the selected Check Box.

[0619] As indicated at Block B in **FIG. 6E1**, this causes a particular type of HTML-encoded document (i.e. called an "HTML form" or "Web input form document") residing on the IPD Server(s) **11"** to be sent to the Web browser of the requesting client subsystem **13** and displayed on the information display frame **21C** thereof (requesting this mode of service). As in the Trademark-Directed Search Mode described above, the HTML form sent in the Product-Description Directed Search Mode may also use any HTML format commands, such as headers, paragraphs, and lists, but must include three unique items, namely: the METHOD by which the user input is to be sent; the ACTION, which specifies a URL to which the user input is to be sent, (e.g. a CGI script running within the IPD server **11"** that will act upon the request appropriately); and a SUBMIT button, to send the completed form over the Internet via HTTP. In the illustrative embodiment, user input (i.e. the description or descriptor for a particular consumer product on which information is sought) is obtained by an Input Box, which allows the user (i.e. retail sales clerk or consumer) to type in the product description for a particular consumer product on which information is sought.

[0620] As indicated at Block C in **FIG. 6E1**, the consumer or retail clerk enters the product description into the Input Box of the HTML form, and selects the SUBMIT button thereon. In response thereto, the Web browser on the client subsystem **13** sends a GET request to the IPD server **11"**.

When selecting the SUBMIT button on the HTML form, the Web browser executes the METHOD associated with the HTML form and sends the stored product description to the URL specified by ACTION associated with the HTML form (i.e. the Web browser performs the action specified in the ACTION). The ACTION of the HTML form specifies the URL of a CGI script within the IPD server **11**" that will process the request from the HTML form. This amounts to the Web browser constructing a GET request for that URL, with the arguments (the query string) being attached to the end of the URL. The arguments of the HTML form are specified by the INPUT items of the HTML form (i.e. the product description for the consumer product on which information is sought).

[0621] As indicated at Block D in **FIG. 6E1**, the httpd program on the IPD server **11**" passes the arguments (the product description in the query string) to a CGI script therewithin and the CGI script translates the Web query string into a proper query to the RDBMS **9** shown in **FIG. 2B2'**.

[0622] As indicated at Block E in **FIG. 6E1**, the translated query is used to search the RDBMS **9** in order to find all registered consumer products having trademarks or tradenames within the RDBMS **9** that are linked to the product description entered into the Input Box of the HTML form. The result returned from the RDBMS **9** is an ASCII record specifying each triplet data set (Trademark, UPN and Manufacturer) which satisfies the above product-description search criteria (to a particular degree) entered into the Input Box of the HTML form. In order for the Web browser of the requesting client subsystem to display the results of the database search during this mode, the ASCII record must be converted into another HTML form for use in refining the consumer product information display.

[0623] At Block F in **FIG. 6E2**, the IPD server **11**" creates the elements of another HTML document (Web auxiliary input form), inserts the preliminary search result from the RDBMS **9** into the Web auxiliary input form, and sets the Content-type of this HTML document to text/html. In the illustrative embodiment, the Web auxiliary-input form has an ACTION which specifies the URL of a CGI script within IPD server **11**" that will act upon the request appropriately as if the system were in the UPN-Directed Information Access Mode. The Web auxiliary input form includes an Input Box listing all triplet data sets (i.e. Trademark, Manufacturer, and UPN number) satisfying the input product-description search criteria entered in the primary Web input document, described hereinabove. The qualifying triplets listed in the Input Box are provided with a Radio-Button to allow the consumer or retail sales clerk to select one of the triplets from the list thereof for use in a subsequent refined search of the RDBMS **9**. The Web auxiliary-input form also has a SUBMIT button for sending the HTML form back to the IPD server **11**" for processing.

[0624] As indicated at Block G in **FIG. 6E2**, when the consumer or retails sales clerk makes a selection with the Radio-Button and then selects the SUBMIT button, the Web browser on the client subsystem **13** sends a request to the IPD server **11**".

[0625] As indicated at Block H in **FIG. 6E2**, the httpd program on the IPD server **11**" passes the arguments (the product description in the query string) to a CGI script therewithin and the CGI script translates the Web query string into a proper query for use in searching RDBMS **9** shown in **FIG. 2B2**.

[0626] At Block I in **FIG. 6E2**, the query is used to search the RDBMS **9** in order to find the set of URLs (i) linked to the registered consumer product (by the manufacturer or agent thereof) assigned the UPN, (Trademark and Manufacturer) entered into the Input Box of the HTML (auxiliary) form, and (ii) pointing to HTML (or FTP) documents on the WWW containing particular types of consumer product related information. The result returned from the RDBMS **9** is an ASCII record specifying the set of URLs satisfying the above search criteria. In order for the Web browser of the requesting client subsystem to display the results of the database search during this mode, the ASCII record must be converted into a HTML document (i.e. output HTML form).

[0627] At Block J in **FIG. 6E2**, the IPD server **11**" creates the elements of an output HTML form, inserts the result from the RDBMS **9** thereinto, and sets the Content-type of this HTML document to text/html and sends a request to the IPD server **11**" to get the HTML form.

[0628] At Block K in **FIG. 6E3**, the set of URLs categorized by particular product information types is displayed within the output HTML form on the information display frame **20C**. Notably, this set of URLs points to particular types of consumer product related information registered within the RDBMS **9** of the system by the manufacturer of the product or its agent(s) thereof using the UPN/URL management tools accessible during the Manufacturer/Product Registration Mode hereof.

[0629] As indicated at Block L in **FIG. 6E3**, the consumer or retail sales clerk can access and display any HTML document (Web page) located at a particular URL within the displayed information menu by selecting the same using a touch screen, mouse, or other input selection device available at the requesting client subsystem **13**.

[0630] The protocols described above can be realized using any suitable programming language including, for example, an object-oriented programming language such as the Java™ programming language.

CPIR-Enabling Applet Download/Distribution Mode of System Operation

[0631] As illustrated in FIGS. **4F1, 4F2, 4H1, 4H2, 4J1, 4J2, 4L1** and **4L2**, a centralized Library of CPIR-enabling Applets/Servlets is created, management and stored within the UPN/URL Database Management Subsystem **9** hereof in accordance with the above-described methods. In accordance with the principles of the present invention, these CPIR-enabling Applets/Servlets must be widely distributed to retailers, manufacturers, advertisers and others about the globe and thereafter widely embedded within HTML-encoded documents, as taught in detail hereinabove, to practice this aspect of the present invention in a commercially successful manner. The function of the CPIR-enabling Applet Download/Distribution mode of operation of the system is to enable the world-wide distribution of this centralized Library of CPIR-enabling Applets/Servlets, in accordance with the licensing program associated with each such CPIR-enabling Applet.

[0632] As shown in **FIG. 3C**, the CPIR-enabling Applet Download/Distribution mode is automatically initiated by

the user depressing mode control button **21F** displayed on the control panel **20B** of the UPC Request (Central or Retail) GUI of the illustrative embodiment, but certainly elsewhere in practice. The user can be anyone with the requisite authority to use the Applets in accordance with the terms of the licensing program to be enforced in connection therewith. Understandably, the terms of such licensing programs will be based on prevailing business conditions and will vary from embodiment to embodiment of the present invention.

[0633] As best illustrated in **FIG. 4F2**, upon entering the UPC-Encoded-Applet-Download/Distribution mode, the IPD server **11** of the illustrative embodiment will serve a custom Java GUI as shown in FIGS. **4M1** and **4M2**, for carrying out Applet tag downloading and licensing procedures. The GUI will provide (1) links to the centralized Library of CPIR-Enabling Applets/Servlets maintained within the UPN/URL Database Management Subsystem **9**, as well as (2) launchable GUIs for downloading selected UPN-identifiable CPIR-enabling Applets to specified Internet-enabled client computer subsystems **13** or Internet information/application servers operated by the user interfacing with this mode of system operation. Notably, electronic data interchange/exchange processes (e.g. based on EDI, XML/ICE or other protocols) can be used to carry out the downloading of CPIR-enabling Applets and other files between client computers and the IPD server **11** during this and other modes of operation.

[0634] For licensing purposes, it may be desirable or necessary to have the user supply "end-use" types of information to the IPD server **11** during this mode of operation in order to identify on which information servers or domains (e.g. www.ipfcorp.com) particular CPIR-enabling Applets are to be used (i.e. embedded within HTML-documents and launched therefrom by the end-user which will typically be the consumer). In some instances, licenses for CPIR-enabling Applets will carry a fee to be paid by the downloader; in other instances, there will be no fee requirements. Such details will depend on prevailing business conditions along the retail supply and demand chain.

[0635] Once a user has downloaded CPIR-enabling Applets onto a designated (target) client machine or server, the user will have built a local Library (or Catalog) of CPIR-enabling Applets for use during Applet embedding operations which will typically be carried out alongside of other HTML-code authoring and management operations involving, for example, the design, construction, management and maintenance of Web-pages, EC-stores, on-line (retail and wholesale) product catalogs, on-line auction site pages, Web advertisements, and the like.

[0636] As shown in **FIG. 4F2**, during Step **D1** of the Applet embedding process, CPIR-enabling requests are accessed from the local Library of CPIR-enabling Applets typically over an IP-type local area network (LAN) or wide area network (WAN). Then during Step **D2**, the accessed CPIR-enabling Applet is inserted within the HTML code of the target document. This step of the process will typically involve use of HTML-editing tools of one sort or another, as discussed herein above.

[0637] Once the CPIR-enabling Applet has been embedded within the target HTML-encoded documents, the HTML-encoded can then be published in its intended publishing environment so that consumers can instantly initiate UPN-directed searches within the centralized UPN/URL Database Management Subsystem **9** hereof by clicking on the CPIR-enabling Applet, and thereafter display the search results within an independent Java GUI which performs the function of a "cyber-kiosk" provided at the consumer's point of presence on the WWW.

[0638] Accessing the Modes of Operation of the IPI Finding System Hereof by launching CPIR-Enabling Applets Embedded Within HTML-Encoded Documents Associated With Electronic-Commerce Enabled Stores and Product Catalogues, Internet Advertisements, On-Line WWW Auction Sites, and the Like

[0639] As described in detail hereinabove, a consumer can automatically produce a CPID-enabling Java GUI (i.e. "cyber-kiosk") by clicking upon a CPIR-enabling Java Applet that has been embedded within the HTML code of any Web-document. As illustrated in FIGS. **4N2**, **4O2**, **4P2**, **4Q2**, **4R2** and **4S2**, each consumer product information display (CPID) enabling GUI of the illustrative embodiment is a new and independent Web browser having a Netscape-type framework, in which the six mode selection buttons **21A** through **21G** of the illustrative embodiment are displayed. When such CPID-enabling GUIs are displayed at the consumer's point of presence on the WWW, the consumer is free to select any one of the mode selection buttons and cause the system to enter the selected mode and precisely deliver the information service associated therewith without disturbing his or her present Cyberspace experience. The description of these modes will be described in detail hereinbelow with reference to FIGS. **2-1**, **2-2**, **2A** and **2C**.

Registration Solicitation Mode of the IPI Finding and Serving Subsystem

[0640] In the illustrative embodiments of the present invention, the data-synchronized IPD Servers of the system hereof **11** are also provided with a n "Automated Registration Solicitation Mode" programmed by the webmaster (or administrator) of the IPI Web-site(s). In this mode, each IPD Server **11** analyzes the data collected within its Non-IPI Registrant Database. The data analysis procedure seeks to determine: (1) which "unregistered" products in the Non-IPI Registrant Database were the subject of an information request at the IPD Server; (2) how many hits (requests) were made for the product within a predetermined length of time (e.g. one week) by Internet users; and (3) whether the number of requests exceeds a particular "request threshold" (e.g. **100** requests in week period). Then for each unregistered product which has exceeded the request threshold, the IPD Server automatically sends an E-mail message to the associated company. Preferably, the E-mail message is designed to (i) inform the company of recent information requests for their products, and (ii) solicit the registration of such products with the IPD Server. Once registered with the system, such products can be easy found on the Internet by anyone wishing to use the product information finding techniques of the present invention.

Operation of the IPI Finding and Serving Subsystem and Method Hereof

[0641] When the Check Box button **21C** is selected from the control frame **20B**, the IPI Finding and Serving Subsystem enters its "UPN-Directed Information Access Mode"

illustrated in **FIG. 6C**. Preferably, the user is provided with a choice of language (e.g. English, German, French, Japanese, Korean, Russian, Chinese, etc.) by way of an appropriate menu-selection screen. After the desired language selection is made, the home page is displayed upon the client subsystem's display screen. A typical display screen produced from the IPD Server might read as follows:

[0642] "Welcome to UPC-REQUEST™, the only Universal Product-Information Finding and Serving System on the Internet.

[0643] Have you purchased a particular product, are you considering the purchase of a particular product, on which you would like current, up-to-date information from the manufacturer or advertiser?

[0644] Look no further than the UPC-REQUEST™ Universal

[0645] Product-Information Finding and Serving System."

[0646] When the subsystem is in its "UPN-Directed Information Access Mode", a Web-based information resource pertaining to any commercial product registered with the system can be displayed and selected by the user in order to automatically access the same from the Internet. Such information resources can include advertisements, specifications, operation descriptions, product simulations, purchase information, maintenance information, warranty and servicing information, product updates, distributor/reseller information, incentives (e.g. discounts, rebates, coupons, etc.), electronic data transaction screens, etc. In this mode, desired product information is obtained by simply entering the registered product's UPN (e.g. its UPC's 12 digit numerical string) into the Inout Box of the HTML form displayed in the information display frame **20C**. Such data entry can be carried out manually using a keyboard data entry techniques, or automatically using a bar code symbol reader connected to the client subsystem as discussed in detail above. When using the seeded IPI Database described hereinabove, only the first six digits of the UPC number need be entered into the dialogue box. An exemplary display screen produced from the IPD Server might be as follows:

[0647] "Simply enter the 12 digit UPC the particular product; click REQUEST, and then wait for the display of the list of Web locators (URLs) at which the desired product information can be found on the Internet?"

[0648] In response to such data entry operations, a list or menu of URLs organized according to information subfield classifications as set forth, for example, in **FIG. 4A2**, are displayed on client subsystem Ca making the request of the IPD Server **11**. At this stage, another display screen associated with the HTML form produced from the IPD Server **11** would appear with an exemplary message as follows:

[0649] "Please select the URL from the displayed URL Menu using the information subfield product information category displayed above. This will connect you to the product information related to the selected URL. You can return to the URL display list at anytime."

[0650] Upon selecting a particular URL from the displayed URL menu, video and audio information content are automatically served from the IPI Server **12** hosting the selected URL and thereafter displayed on the client subsystem **13**.

[0651] When the Check Box button **21D** is selected, the IPI Finding and Searching Subsystem enters its Trademark-Directed Search Mode, illustrated in FIGS. **6D1** through **6D3**. Preferably, the user is provided with a choice of language (e.g. English, German, French, Japanese, Chinese, etc.) by way of an appropriate menu-selection screen.

[0652] When the system is in its Trademark-Directed Search Mode, a predesignated information resource pertaining to any commercial product registered with the system can be automatically accessed from the Internet and displayed from the Internet browser of a client subsystem **13**. Such information resources can include advertisements, specifications, operation descriptions, product simulations, product upgrade information, purchase information, maintenance information, warranty and servicing information, etc. In this mode, desired product information is obtained by simply entering the registered product's trademark(s) and/or associated company name into the Input Box of the HTML form displayed on the information display frame **20C** of the client subsystem. An exemplary message associated with the HTML form produced from the IPD Server **11** might be as follows:

[0653] "Simply enter the trademark used in connection with the particular product and/or the company name of the product's manufacturer; click REQUEST, and then wait for the display of a list of Web locators (URLs) at which desired types of product information can be found on the Internet"

[0654] In response to such data entry operations, a list of URLs organized according to the information subfield classifications set forth in **FIG. 4A2** are displayed on client subsystem placing the request. Upon selecting a particular URL from the displayed list thereof, video and audio information content are automatically served from the IPI Server hosting the selected URL and thereafter displayed on the client subsystem.

[0655] In an alternative embodiment of the present invention, the UPN-Directed Information Access Mode and the Trademark-Directed Search Mode can be integrated into a single server application so that there is no need or desire to manually select between mode activation buttons **21C** and **21D**, respectively. In such an embodiment, the interaction between the IPD Server and the requesting client subsystem can be designed to support the following Web server display screens and script underlying the same:

[0656] "Welcome to UPC-REQUEST™, the only Universal Product-Information Finding and Serving System on the Internet.

[0657] Have you purchased a particular product, or considering the purchase of a particular product, on which you would like current, up-to-date information from the manufacturer or advertiser?

[0658] "Look no further than the UPC-REQUEST™ Universal Product Information Finding and Serving System."

[0659] "Simply enter the 12 digit UPC number of the particular product, click REQUEST, and select from

the displayed menu of Web locators (URLs) to find the desired product information on the WWW.

[0660] "If you do not know the UPC number associated with the product you are looking for, then simply enter the trademark used in connection with the particular product and/or the company name of the manufacturer, then click REQUEST, and wait for the display of the list of Web locators (URLs) at which the desired product information can be found.

[0661] "Please select the URL from the displayed URL list by clicking on it. This will connect you to the product information related to the selected URL You can return to the URL display list at anytime."

[0662] Notably, such an integrated Web server application can be realized in a variety of ways. The exact words and graphics used to create an interactive script for an integrated Web server application will vary from embodiment to embodiment of the present invention.

[0663] In instances when an IPI Website in accordance with the present invention is being served to consumers in retail environments using a computer-based kiosk as shown in **FIG. 3A2**, the consumer as well as retail sales clerk is presented with the option of ascertaining the price of an product in the store. This is achieved by simply depressing the "Price Display" button **21F** on Control Strip **20B**, shown in **FIG. 3C**, to engage the system in its price lookup/display mode. In this mode of operation, the consumer then need only scan the UPC bar code symbol on the product using bar code scanner **26** in order for the price to be looked-up in the Product Price Database maintained in the Retailer's Price Server (RPS) **35**, and displayed on the kiosk display screen. In general, the Product Price Database of the hosting retailer can be made accessible by the computer-based kiosk in several possible ways. As shown in FIGS. **2-1** and **2-2**, one way is to place the retailer's RPS on Internet (by using an HTTP server) and connect the RPS to the IPD Server **11** of the system by way of a CGI well known in the art. The CGI can be made accessible only by authorized client subsystems (e.g. computer-based kiosks installed in the hosting retailer's store and possibly administrators of the information delivery system). An alternative technique of connecting the Product Price Database t o each computer-based kiosk would involve providing the RPS with a direct interface to each computer-based kiosk in the hosting retailer's store(s). This alternative technique may require the use of computer networking technology well known in the art.

[0664] Method of Accessing and Displaying and Consumer Product Related Information Within Retail Shopping Environments and Transporting the Same to Remote Locations for Subsequent use and/or Review by Consumers at Home, Work, School or on the Road

[0665] As explained hereinabove, the problem of providing consumers with copies of accessed consumer product information within retail shopping environments is addressed by enabling the consumer at retail-based bar code driven kiosks to: (1) display an e-mail envelope within the display frame **20C** of the Web browser program thereof, by manually selecting control button **21G** provided along the control frame **20B**, shown, for example, in FIGS. **3A14A** and **3A14B**; (2) capturing, saving, and attaching any accessed/displayed consumer product document to the displayed e-mail envelope **116** by manual selection of the "capture, save and attach" button **110** (or voiced-directed selection thereof) within the displayed e-mail envelope of **FIG. 3A14**, or capturing and recording the URL of the CPI-related document being displayed by manual selection of the "capture and record" button **112** (or voice-directed selection thereof) within the displayed e-mail envelope **116** of **FIG. 3A16B**; (3) addressing the e-mail envelope **116** with the consumer/shopper's home, office or like e-mail address by either reading an e-mail address encoded within a bar code (or magnetic-stripe) structure or manually entering the same within the addressee field **115**; and (4) sending the stuffed e-mail envelope by manual selection of the "send" button **114** within the displayed e-mail envelope.

[0666] At this stage, the first illustrative embodiment of the CPI transport method (i.e. service) of the present invention referred for hereinafter as the "SEND-IT-HOME™" e-mail service will now be described with reference to FIGS. **3413A** through **3A14**.

[0667] As indicated at Block A in **FIG. 3A13A**, the first step of the first illustrative embodiment of the CPI transport method involves launching a consumer product information (CPI) capture and transport application/service on a n Internet-enabled bar code driven (BCD) CPI kiosk of the present invention, as shown in FIGS. **3A9** through **2A10D**. As indicated in **FIG. 3A14**, this application launching process can be initiated by selecting application/service button **21G** in the control strip of the kiosk browser display screen. When the application has been launched, a CPI-transporting "electronic-mail" envelope **116** will appear within the display frame of the browser's GUI, as shown in **FIG. 3A14**. As shown therein, the (Java-enabled) GUI for the CPI transporting e-mail envelope is provided with:

[0668] (i) a first single-click button **110** for capturing and storing thereon, as an HTML-encoded document, any CPI related document that is being displayed on the display screen of the BCD CPI kiosk **13** within the retail shopping environment, shown in FIGS. **3A9** through **3A10D**;

[0669] (ii) a second single-click button **114** for transporting copies of the envelope **116** to the e-mail address of the consumer/shopper or friend/agent thereof;

[0670] (iii) a consumer e-mail address field **115** for entering the e-mail address of the consumer/shopper or friend thereof, to which a copy of the e-mail envelope **116** can be automatically sent during envelope transport; and

[0671] (iv) a retailer e-mail address field **117** containing a preset e-mail address of the retailer operating the kiosk, indicating the retail store location, and possibly the retail department from which the CPI-transporting envelope **116** was sent on the time and date of the electronic message transmission.

[0672] As indicated at Block B in **FIG. 3A13A**, the consumer enters into the consumer e-mail address field of the envelope **116** , either his or her e-mail address or the e-mail address of a friend to which he or she would like to send consumer product related information. Notably, this operation can be achieved in a variety of different ways,

namely: by manually typing the e-mail address using a pop-up keyboard or keypad provided for by the kiosk itself; using voice recognition command technology embodied within the kiosk; by reading a bar coded consumer identification card **150**, as shown in **FIG. 3A10B**, having the consumer's e-mail address and possibly other information items (e.g. shopper/consumer identification number, credit card information, name, address and/or status within a particular loyalty/courtesy program encoded therewithin if desired; or, by reading a magnetic-stripe type consumer identification card **150** encoded with the same or similar information using a magnetic stripe reader **46** interfaced with the BCD CPI kiosk, as shown in **FIG. 3A10B**.

[0673] As indicated at Block C in **FIG. 3A13B**, the consumer/shopper uses the UPN, trademark, product descriptor and/or company name associated with the sought after product, to access consumer product related information resource of interest on the WWW, and display the same on the display screen of the BCD CPI kiosk within the retail shopping environment. When using the UPN-Directed Search Mode of operation of the system, a UPN/URL menu as shown in FIGS. **4P2**, **4R2** and **452**, will be displayed in the display frame of the browser screen. When using the Trademark Directed Search Mode, a list of URLs linked to the input trademark will be displayed in the display frame of browser screen.

[0674] As indicated at Block D in **FIG. 3A13B**, each instant the consumer/shopper has found a consumer product information resource of particular interest on the WWW, which he or she wants to send a copy thereof to a home or work e-mail address, the consumer/shopper selects the first single-click button **110** on the CPI-transporting envelope (indicated as "CAPTURE, SAVE & ATTACH" in **FIG. 3A14**). This causes the information resource being displayed on the display screen of the BCD CPI kiosk to be automatically captured and stored as an HTML-encoded document attached to the CPI-enabling envelope, as intended in **FIG. 3A14**.

[0675] As indicated at Block E in **FIG. 3A13B**, the consumer/shopper sequentially repeats Steps C and D for each consumer product information resource on the WWW to be captured, stored and attached to the opened CPI-transporting envelope **116** being displayed on the display screen of the kiosk. Typically, there will be a maximum number of documents to be attached to the CPI-enabling envelope using present e-mail protocols at 6 understood, however, that a larger number of documents may be transportable within a single e-mail envelope using future 3-mail protocols.

[0676] As indicated at Block F in **FIG. 3A13C**, upon capturing, storing and attaching a desired number of consumer product related information resources to the CPI-transporting envelope **116**, the consumer/shopper then selects the second single-click button **114** on the CPI-transporting envelope (i.e. indicated as "SEND" in **FIG. 3A14**) so as to transport the CPI-transporting envelope **116** to the e-mail address entered within the consumer e-mail address field of the CPI-transporting envelope.

[0677] As indicated at Block G in **FIG. 3A13C**, an information record of each CPI-transporting envelope and the contents thereof is maintained at the retailer e-mail server **84** and a copy thereof is transported to the central e-mail server **88**, shown in FIGS. **3A9** and **3A10**. Thereafter, such information on the central e-mail server **88** can be analyzed to determine trends and patterns in consumer shopping behavior in different geographic locations and retail shopping departments and the like. The results of such analysis are stored in retailer and manufacturer RDBMSs **88A** and **89B** , respectively.

[0678] The second illustrative embodiment of the CPI transport method (i.e. service) of the present invention, referred to hereinafter as the SEND-IT-HOME™ e-mail service, will now be described with reference to FIGS. **3A15A** through **3A1G**.

[0679] As indicated at Block A in **FIG. 3A15A**, the first step of the second illustrative embodiment of the CPI transport method involves launching a consumer product information (CPI) capture and transport application/service on an Internet-enabled bar code driven (BCD) CPI kiosk of the present invention, as shown in FIGS. **3A9** through **3A10D**. As indicated in **FIG. 3A16**, this application launching process can be initiated by selecting application/service button **210** in the control strip of the kiosk browser display screen. When the application has been launched, a CPI-transporting "electronic-mail" envelope will appear within the display frame of the browser's GUI, as shown in **FIG. 3A16**. As shown therein, the (Java-enabled) GUI for the CPI transporting e-mail envelope is provided with:

[0680] (i) a first single-click button **110** for capturing and storing therein, as an HTML-encoded document, any CPI related document that is being displayed on the display screen of the BCD CPI kiosk **13** within the retail shopping environment, as shown in FIGS. **3A9** through **3A10D**;

[0681] (ii) a second single-click button **114** for transporting copies of the envelope **116** to the e-mail address of the consumer/shopper or friend/agent thereof;

[0682] (iii) a consumer e-mail address field **115** for entering the e-mail address of the consumer/shopper, to which a copy of the e-mail envelope **116** can be automatically sent during envelope transport;

[0683] (iv) a retailer e-mail address field containing a preset e-mail address of the retailer operating the kiosk, indicating retailer store location, and possibly retailer department from which the CPI-transporting envelope **116** was sent on the time and date of the electronic message transmission; and

[0684] (v) a message field **119** to record captured URLs, as well as other notes of the consumer.

[0685] As indicated at Block B in **FIG. 3A15A**, the consumer enters his or her e-mail address into the consumer e-mail address field of the envelope **116**, his or her e-mail address or the e-mail address of a friend to which he or she would like to send consumer product related information. Notably, this operation can be achieved in a variety of different ways, namely: by manually typing the e-mail address using a pop-up keyboard or keypad provided for by the kiosk itself; using voice recognition command technology; by reading a bar coded consumer identification card **150**, as shown in **FIG. 3A10B**, having the consumer's e-mail address and possibly other information items (e.g.

shopper/consumer identification number, credit card information, name, address, and/or status within a particular retailer loyalty/country program) encoded therewithin if desired; or by reading a magnetic-stripe type consumer identification card **150'** encoded with the same or similar information using a magnetic stripe reader **46** interfaced with the BCD CPI kiosk, as shown in **FIG. 3A10B**.

[0686] As indicated at Block C in **FIG. 3A15B**, the consumer/shopper uses the UPN, trademark, product descriptor and/or company name associated with the sought after product, to access a consumer product related information resource of interest on the WWW, and display the same on the display screen of the BCD CPI kiosk within the retail shopping environment. When using the UPN-Directed Search mode of operation of the system, a UPN/URL menu, as shown in FIGS. **4D2**, **4R2** and **4S2**, will be displayed in the display frame of the browser screen. When using the Trademark Directed Search mode, a list of URLs linked to the input trademark, as shown in **FIG. 4T2**, will be displayed in the display frame of browser screen.

[0687] As indicated at Block D in **FIG. 3A15B**, each instant the consumer/shopper has found a consumer product information resource of particular interest, which he or she wants to send the corresponding URL to a home or work e-mail address, the consumer/shopper selects the first single-click button **110** on the CPI-transporting envelope **116'** (indicated as "CAPTURE AND RECORD" in **FIG. 3A16**). This causes the URL of the information resource being displayed on the display screen of the BCD CPI kiosk to be automatically captured and recorded within the message field **119** of the CPI-enabling envelope **116"**. Optionally, the consumer may type a short note in this field using a "pop-up" keyboard launched by "pop-up" keypad button **121** on envelope **116'**.

[0688] As indicated at Block E in **FIG. 3A15B**, the consumer/shopper sequentially repeats Steps C and D for each consumer product information related URL to be captured and recorded within the message field **119** of the opened CPI-transporting envelope **116'**.

[0689] As indicated at Block F in **FIG. 3A15C**, upon capturing and recording a desired number of consumer product related URLs to the CPI-transporting envelope **116'**, the consumer/shopper then selects the second single-click button **114** on the CPI-transporting envelope (i.e. indicated as "SEND" in **FIG. 3A16**) so as to transport the CPI-transporting envelope to the e-mail address entered within the consumer e-mail address field **115** of the CPI-transporting envelope.

[0690] As indicated at Block G in **FIG. 3A15C**, an information record of each CPI-transporting envelope and the contents thereof is maintained at the retailer e-mail server **84** and a copy thereof is automatically transported to the central e-mail server **88**. Thereafter, such information on the central e-mail server **88** can be analyzed to determine trends and patterns in consumer shopping behavior in different geographic locations and retail shopping departments and the like. The results of such analysis are stored in retailer and manufacturer RDBMSs **89A** and **89B**, respectively.

[0691] The above-described methods of e-mail based CPI transport may be modified in various ways to satisfy particular requirements of the application at hand. Also, these methods may be modified to provide novel ways of transporting displayed CPI-related Web documents to remote e-mail addresses while a consumer/shopper is visiting an E-commerce enabled store or on-line product catalog.

A Best Mode Embodiment of IPI Finding and Serving Subsystem of the Present Invention

[0692] The IPI finding system (i.e. subsystem) of the present invention can be realized on the Internet in a variety of different ways. Each embodiment of the system will provide manufacturers, retailers, consumers and sponsors with various benefits hitherto unachievable using prior art systems and methodologies. A best mode embodiment of the system will now be described below with reference to Intent-to-Use (ITU) servicemarks that Assignee hereof has sought to register under the Lanham Act and ultimately use in reducing the Internet-based consumer product information finding system to commercial practice in the near future. The inventive subject matter herein disclosed can be readily applied to carry out such an Internet-based information finding and delivery system.

[0693] According to the best mode embodiment, the IPI Finding and Serving Subsystem **2**, referred to hereinabove as the "UPC REQUEST™ Consumer Product Information Finding System" in **FIGS. 7 and 8** hereof, comprises an integration of several subsystems including, for example: the UPC REQUEST™ Manufacturer/Product Registration Subsystem **33** (e.g. Web Document Server **30** and Workstation **31**) including Web-based and Value Added Networks (VAN)-based infrastructure and processes **14** for supporting EDI and UPN/URL database management operations by manufacturers and/or their agents; the UPC REQUEST™ Database Management Subsystem **9** interfaced with the UPC REQUEST™ Manufacturer/Product Registration Subsystem **33**; numerous UPC REQUEST™ kiosks (e.g. client subsystems **13**) installed in retail stores, retail outlets and the like, each having a bar code symbol driven Internet browser providing access to the Internet through an Internet Service Provider (ISP); and all of the Web-enabled client subsystems **13** located in consumer homes, in consumer offices and on the road, having access to the Internet through an ISP. While distributed geographically, these subsystems are integrated through the infrastructure of the Internet.

[0694] The function of the UPC REQUEST™ Manufacturer/Product Registration Subsystem **33** is two-fold: (1) to enable qualified manufacturers to quickly and easily register their companies with the System (i.e. the UPC REQUEST™ Database Management Subsystem **9**) by way of a Web-enabled computer system of their choice; and (2) to enable manufacturers and/or their agents to (i) easily link, manage and update their UPC numbers and linked URLs using any Web-enabled computer system **13** running the EDI (or XML/EDI) based UPN/URL Database Management software (downloaded during manufacturer registration), and periodically transmit such updated information to the UPC REQUEST™ Database Management Subsystem in order to update each manufacturer's information within the UPC REQUEST™ Database (i.e., IPI Database shown in FIGS. **4A1** and **4A2**).

[0695] The function of the UPC REQUEST™ Database Management Subsystem **9** is to maintain and update the UPC REQUEST™ Database (shown in FIGS. **4A1**, **4A2** and

FIGS. 4C through 4C4), which contains various information items regarding registered manufacturers, service-subscribing retailers, and registered consumer products including, for example, UPC (and/or UPC/EAN) numbers assigned to consumer products and linked URLs pointing to published HTTP-encoded documents (i.e. Web pages) containing particular types of information related to such products.

[0696] Within the store of each retailer subscribing to the UPC REQUEST™ Consumer Information Service, the function of the UPC REQUEST™ kiosk is to provide consumer access to the UPC REQUEST" Retailer Website (e.g. UPC REQUEST" Retail @ Wal-Mart, UPC REQUEST™ Retail @ Home Depot, etc.). The UPC REQUEST™ Retailer Website served to both physical-kiosk and cyber-kiosks within the retailer's brick and mortar and EC stores, respectively, provides consumer access to UPN/URL information links (i) relating only to those products sold by the retailer and maintained within the UPC REQUEST™ Database Management System by the manufacturer or agent thereof (i.e. achieved by applying MIN-based UPN-filtering techniques to the UPN/URL database, and (ii) not containing distribution channel information. If desired by the subscribing retailer, its UPC REQUEST™ Retailer Website can be freely served to customers over the Internet, e.g. accessible from a hot-link embedded somewhere in the retailer's Web-site.

[0697] Within the realm of the UPC REQUEST™ System **2**, the function of the Web-enabled client computer system **13** of each consumer, wherever it may be located (e.g. at home, in the office or on the road), is to provide consumer access the UPC REQUEST™ Central Website which is freely served over the Internet to any consumer having a Web-enabled computer system. Unlike each UPC REQUEST™ Retailer Website maintained by the UPC REQUEST™ Database Management System, the UPC REQUEST™ Central Website provides consumer access to UPN/URL information links relating to every product maintained within the UPC REQUEST™ Database Management System by every registered manufacturer. Any attempt by a consumer to access information from a particular UPC REQUEST™ Retailer Website regarding a product not sold in the retailer's store will automatically result in a link over to the UPC REQUEST™ Central Website.

[0698] Depending on the particular implementation of the system hereof, it might be desirable or necessary for particular or all UPC Request™ Central Website(s) to be designed so that is does no support "Web-clipping" or functionally equivalent techniques so that hand-held wireless (client) computing devices **13**, whether or not equipped with a bar code symbol reader, and free to move anywhere including Retail stores subscribing to UPC Request Retailer WWW site services, cannot receive "Web-Clipped" HTML-encoded pages served from UPC Request™ Central WWW Site, Retailers subscribing to UPC Request Retailer WWW sites can be confident the consumers/shoppers cannot access UPC Request™ Central WWW Service within retail store environmentally using wireless web-enabled hand-held devices having display panels substantially smaller than the dimensions of web pages being served therefrom.

A Brief Description of the UPC REQUEST™ Information Service Suite

[0699] When installed in retail stores, the UPC REQUEST™ System provide will provide seven revolutionary Internet-based consumer information services under the servicemarks HOME-PAGE™, INFO-LINK™, CYBER-SERVICE™, TRADEMARK™, PRODUCT-TYPE™, UPC-ENCODED-APPLE-DOWNLOAD, and SEND-IT-HOME™, respectively. Each of these information services is accessible to consumers and sales clerks alike from a UPC REQUEST™ Retailer Website (e.g. UPC REQUEST™@ Home Depot Website) accessed within a retail store, as well as from on the UPC REQUEST™ Central Website.

[0700] In the UPC REQUEST™ version of the IPI Finding and Serving Subsystem of the present invention, the INFO-LINK™ provision is supported during and enabled by Manufacturer/Product Registration Mode; the HOME-PAGE™ provision is supported during and enabled by the Manufacturer Website Search Mode; the CYBER-SERVICE™ provision is supported during and enabled by UPN-Directed Information Access Mode; TRADEMARK™ provision is supported during and enabled by the Trademark-Directed Search Mode; PRODUCT-TYPE™ provision is supported during and enabled by the Product-Description Directed Search Mode; and UPC-ENCODED-APPLET-TAG-DOWNLOAD/DISTRIBUTE(, provision is supported during and enabled by the UPC-Encoded Applet Tag Download/Distribution Mode. Each of these system modes have been described in great detail hereinabove.

[0701] To constantly remind the public at large of the "fee-paying" sponsors of the UPC REQUEST™ System, all Web pages displayed by the UPC REQUEST™ System in a retail store (e.g. on UPC REQUEST™@ Home Depot Website), or on the UPC REQUEST™ Central Website, will be displayed within a three-frame display "framework" comprising a sponsor frame, a control frame, and a n information frame.

[0702] The "sponsor frame", located on the upper-most portion of the Internet browser screen, displays the sponsor's greeting such as, for example, "Welcome to UPC REQUEST, sponsored by Visa and Federal Express."

[0703] The "control frame", located on the left-most side of the Internet browser Is screen, will provide six mode activation buttons. The first mode activation button **21A** enables consumers to request the INFO-LINK™ service. The second mode activation button **21B** enables consumers to request the HOME-PAGE™ service. The third mode activation button **21C** enables consumers to request CYBER-SERVICE. The fourth mode activation button **21D** enables consumers to request TRADE-MARK™. The fifth mode activation button **21E** enables consumers to request PRODUCT-TYPE™. The sixth mode activation button **21F** enables consumers to request UPC-ENCODED-APPLET-DOWNLOAD/DISTRIBUTE™. A seventh button **21G** enables the launching of the SEND-IT-HOME™ e-mail transport service of the present invention. An additional button can be provided to enable the download a free plug-in software module which automatically installs a "Product Information" button on the graphical user interface of the consumer's Internet browser, so that the UPC REQUEST™ Central Website can be accessed anywhere in the world with a single click of the mouse button on any pointing device.

[0704] The "information frame", occupying the balance of the Internet browser screen, will display: all HTTP (i.e. Web) and FTP pages launched by in-store scanning of

UPC-labeled products during HOME-PAGE™ or CYBER-SERVICE™; all HTTP and FTP pages launched by clicking on hypertext-links embedded within Web pages accessed through a particular UPC REQUEST™ Retailer Website in retail stores or from the UPC REQUEST™ Central Website; as well as all information search and display (menu) screens served by a UPC REQUEST™ Retailer Website to the UPC REQUEST™ Central Website.

[0705] While the web-enabled kiosks of the present invention **13** have been shown herein as employing a commercial-type Internet browser program (i.e. http client program), preferably with instructions on browsing in retail shopping environments, it is understood that the browser programs employed in such kiosks can be customized for each retailer in whose store the kiosk is installed, and could be provided with "minimal" browser control buttons (e.g.

[0706] Page Forward →, Page Backward ←, Stop ⊗, and Reload), which, in cooperation with the control buttons in the control frame **20B**, provide a suitable GUI for use in both brick and mortar and electronic retail shopping environments.

[0707] In retail stores subscribing to the UPC REQUEST™ System, HOMEPAGE™ manufacturer's Website search site will enable consumers to automatically access the WWW Home Page of any registered manufacturer by scanning the UPC (or UPC/EAN) bar code symbol on any product thereof using the bar code symbol reader associated with a UPC REQUEST™ kiosk. In general, the UPC REQUEST™ kiosk can be realized by any Web-enabled computer system **13** having an Internet browser program, on-line access to the UPC REQUEST™ Retailer Website, and optionally a touch-screen display panel. The UPC REQUEST™ kiosk may, however, be realized as an inexpensive Internet access terminal comprising a Web-enabled network computer (NC), an LCD touchscreen panel, and a laser scanning bar code symbol reader integrated within a n ultra-compact housing that is mountable within diverse locations within retail stores. As shown in **FIG. 3A5**, the UPC REQUEST™ kiosk may also be integrated within a conventional Point Of Sale (POS) station having a laser scanning bar code symbol reader and a large rotatable LCD display panel. Being as easy to install as a telephone modem, UPC REQUEST™ kiosks of this design can be widely deployed throughout retail stores world-wide with minimal modifications to the preexisting information infrastructure, and illustrated hereinabove.

[0708] At home, in the office, or on the road, HOME-PAGE™ enables consumers to automatically access the WWW Home Page of any registered manufacturer **10** by entering the UPN (or UPC/EAN number) on any product into the search screen served up by a particular UPC REQUEST™ Retailer Website, or by the UPC REQUEST™ Central Website.

[0709] INFO-LINK™, carried out using EDI (or XML/EDI) based UPN/URL Database .Management software (downloaded from Manufacturer/Product Registration Subsystem **33**), enables manufacturers to simply relate (link), manage and update therein (i) the UPN (or UPC/EAN number) on any product with (ii) the Internet address (i.e. URL) of product-related Web pages published on the Internet by the manufacturer, its agents, or others, for subsequent access and display by consumers using CYBER-SERVICE™. While the IFO-LINK™ service would be made accessible through UPC REQUEST™ kiosks **13** in retail stores (i.e. for the sake of vendors who frequent the same), the actual UPN/URL information linking and management operations associated with the INFO-LINK™ service will typically occur in the "back-offices" of registered manufacturers using Internet-enabled computer systems accessing INFO-LINK™ through a hot-linked URL posted on the UPC REQUEST™ Retailer Website and/or the UPC REQUEST™ Central Website.

[0710] UPC-ENCODED-APPLET-DOWNLOAD™, carried out using EDI (or XML/EDI) based processes, enables manufacturers (via the system administrator) to distribute CPIR-enabling Applets/Servlets to retailers, manufacturers, advertisers and others about the globe so that they may embed the same within HTML-encoded documents in order that consumers can instantly initiate single mouse-click UPN-directed, Trademark-directed and/or Product-Descriptor-directed CPI searches within the centralized UPN/URL Database Management Subsystem **9** hereof and display the search results within a CPID-enabling Java GUI which performs the function of a "cyber-kiosk" provided at the consumer's point of presence on the WWW. In the context of the illustrative embodiment of the system of the present invention, such single mouse-click initiated CPI searches solve a major fear of most retailers in both Physical and Cyber Space retail environments, namely: getting a customer into their store, and then having them step out to get some more advice, information or endorsement before making a purchase, or worse yet, never returning to make a purchase, and instead shopping elsewhere for the sought after product.

[0711] CYBER-SERVICE™, accessible through a particular UPC REQUEST™ Retail Website or the UPC REQUEST™ Central Website, enables consumers at home, in the office, on the road, and in retail stores, to quickly access particular types of product-related information which have been published on the WWW by registered manufacturers, their agents and others about consumer products registered with the UPC REQUEST™ Database Management System. CYBER-SERVICE™ displays such product-related information in a menu-like format organized by particular information types (e.g., Product Advertisements, Product Endorsements, Product Reviews, Product Rebates and Incentives, Product Description, Product Manual/Instructions, Product Updates (at FTP Sites), Product Returns, Warranty and Repair Service, Direct Product Purchase, Retailers, Wholesalers, Complementary Products, Company Annual Report, Stock Purchase, etc.). Each displayed information menu associated with a consumer product contains hyper-linked URLs pointing to HTML-documents containing particular types of product-related information linked to the product by the manufacturer or its agent.

[0712] When CYBER-SERVICE™ is selected from a particular UPC REQUEST™ Retail Website, each product-related Web page listed in the displayed "information menu" can be accessed and displayed simply by touching the corresponding Internet address (i.e. URL) displayed on the touch-screen display panel of the UPC REQUEST™ kiosk in the retail store. When CYBER-SERVICE™ is selected from the UPC REQUEST™ Central Website, each product-related Web page listed in the displayed "information menu"

can be accessed and displayed simply by clicking the display screen thereof accessed by an Internet-enabled computer system.

[0713] TRADE-MARK™, accessible through a particular UPC REQUEST™ Retail Website or the UPC REQUEST™ Central Website, enables consumers to quickly access particular types of product-related information from the UPC REQUEST™ Database, by using the trademark or tradename of the related product. When this mode of service is requested, a search screen is displayed within the information frame so that the consumer or sales clerk can enter the trademark or tradename for the related consumer product.

[0714] PRODUCT-TYPE™, accessible through a particular UPC REQUEST™ Retail Website or the UPC REQUEST™ Central Website, enables consumers to quickly access particular types of product-related information from the UPC REQUEST™ Database, by using a descriptive term for the related product. When this mode of service is requested, a search screen is displayed within the information frame so that the consumer or sales clerk can enter a descriptive term for the related consumer product.

[0715] SEND-IT-HOME™ e-mail service, accessible through a particular UPC REQUEST™ Retail Website or UPC REQUEST™ Central Website, enables a consumer to send to a desired e-mail address at home, work or elsewhere, consumer product information on the WWW, accessed from a Web/e-mail-enabled kiosk in a retail shopping environment.

[0716] To maximize value to a particular retail store's customers, each UPC REQUEST™ Retail Website served at each retail store subscriber would be made accessible to consumers outside their retail stores (e.g. at home, in the office or on the road) by several Internet access methods including, for example: through a hot-linked URL posted on the retail store's Website, pointing to the UPC REQUEST™ Retail Website; through a publicly accessible URL, e.g. http://www.upcrequest.com/ @ retail store; etc.

Benefits Provided to Sponsors Promoting the UPC REQUEST™ System

[0717] HOME-PAGE™ and CYBER-SERVICE™ will foster customer loyalty and patronage with sponsors by providing consumers with valuable product-related information before and after product purchases.

[0718] CYBER-SERVICE™ will provide sponsors with an economical and effective way of marketing products and services to their customers and shoppers worldwide.

[0719] HOME-PAGE™ will provide sponsors with valuable advertising space on the UPC REQUEST™ Central Website, for subleasing to retailers, manufacturers and business concerns.

[0720] INFO-LINK™ and UPC-ENCODED-APPLET-DOWNLOAD/DISTRIBUTE will provide sponsors with an effective way of promoting their products and/or services among manufacturers, retailers and consumers using the UPC REQUEST™ System.

Benefits Provided to Customers Using the UPC REQUEST™ System in Retail Stores

[0721] HOME-PAGE™ will allow customers to automatically access Home Page of any manufacturer's Website by simply scanning the UPC bar code symbol on any product thereof at any UPC REQUEST™ information kiosk in a retailer's store.

[0722] CYBERSERVICE™ will allow customers to quickly access (by touch-screen URL selection) particular types of product-related information that have been published on the World Wide Web (WWW) by manufacturers, their agents or others, and registered within the UPC REQUEST™ Database through INFO-LINK™.

[0723] TRADE-MARK™ will allow customers to quickly access product related information menus from the UPC REQUEST™ Database using trademarks or tradenames used in connection with the product on which information is being sought.

[0724] PRODUCT-TYPE™ will allow customers to quickly access product related information menus from the UPC REQUEST™ Database using descriptive terms related to the product on which information is being sought.

[0725] SEND-IT-HOME™ will allow customers to simply transport home, to the office or elsewhere, consumer product information accessed from Web-enabled kiosks installed in retail shopping environments.

Benefits Provided to Customers Using the UPC REQUEST™ System at Home, in the Office, or on the Road

[0726] HOME-PAGE™ will allow customers to automatically access the Home Page of any manufacturer's Website by simply entering the UPC (or UPC/EAN) number on any consumer product, into a UPC REQUEST™ search screen served from the UPC REQUEST™ Central Website and accessible to anyone using an Internet-enabled computer system at home, in the office, or on the road.

[0727] CYBER-SERVICE™ will allow customers to quickly access from the UPC REQUEST™ Database, particular types of product-related information that have been published on the WWW by manufacturers, their agents or others, and registered within the UPC REQUEST™ Database through the INFO-LINK™ service.

[0728] TRADE-MARK™ will allow customers to quickly access product related information menus from the UPC REQUEST™ Database using trademarks or tradenames used in connection with the product on which information is being sought.

[0729] PRODUCT-TYPE™ will allow customers to quickly access product related information menus from the UPC REQUEST™ Database using descriptive terms related to the product on which information is being sought.

Benefits Provided to Retailers Providing UPC REQUEST™ System in Their Stores

[0730] HOME-PAGE™ and CYBERSERVICE™ will foster customer loyalty and patronage with retailers by providing shoppers with valuable product-related information before and after product purchases.

[0731] CYBER-SERVICE™ will provide retailers with an economical and effective way of marketing its products to customers and shoppers within its stores.

[0732] CYBER-SERVICE™ will provide retail sales personnel with an opportunity to learn about a retailer's products by scanning the UPC bar code symbols on such products, and selecting product-related Web pages for in-store review and sales-training sessions whenever customer demand allows.

[0733] HOME-PAGE™ and CYBER-SERVICE™ will provide retailers with valuable advertising space in their stores for subleasing to manufacturers and others (e.g. local sponsors) in order to self-finance the store-wide delivery of the UPC REQUEST™ Service.

[0734] TRADE-MARK™ will allow sales clerks and customers to quickly access product related information menus from the UPC REQUEST™ Database using trademarks or tradenames used in connection with the product on which information is being sought.

[0735] PRODUCT-TYPE™ will allow sales clerks and customers to quickly access product related information menus from the UPC REQUEST™ Database using descriptive terms related to the product on which information is being sought SEND-IT-HOME™ will allow retail sales clerks to send information requests on products to product manufacturers.

Proposed Revenue Model for the UPC REQUEST™ System

[0736] The UPC REQUEST™ System will generate revenue from at least four sources: Manufacturer Fees paid by manufacturers who want to register their products and product-related Web pages with the UPC REQUEST™ Database; Retailer Fees paid by retailers who want to provide bar code driven access to the UPC REQUEST™ System in retail stores; Sponsor Fees paid by Sponsors of the UPC REQUEST™ System; and Advertiser Fees paid by advertisers on the UPC REQUEST™ Central Website.

[0737] Manufacturers desiring to register their consumer products and product-related Web pages within the UPC REQUEST™ Database would pay a one-time Manufacture Registration Fee, based on volume of sales. An annual maintenance fee may be desired or necessary. Minimally, such fees should cover the cost of the EDI and UPN/URL Database Management software (and updates) automatically downloaded to each manufacturer upon registration. Such Web-based EDI software enables manufacturers to easily manage the UPC numbers and Web page URLs associated with their changing product lines, and automatically transmit such information to the UPC REQUEST™ Manufacturer/Product Registration Subsystem in order that the UPC REQUEST™ Database Management Subsystem is periodically updated.

[0738] Retailers providing bar code driven access to the UPC REQUEST™ System in their retail stores would pay an annual Retailer License Fee based on the number of UPC REQUEST™ kiosks deployed therein (within store isles, at POS stations and behind information/service counters).

[0739] Each sponsor of the UPC REQUEST™ System would pay an annual Sponsor Fee for the right to display its name, trademark/servicemark and/or message in hypertext within a selected portion of the sponsor frame displayed by licensed UPC REQUEST™ kiosks in retail stores, as well as Internet-enabled computer systems accessing the UPC REQUEST™ Central Website. For marketing reasons, it would be desirable to limit the number of sponsors of the UPC REQUEST™ System at any instant in time.

[0740] Advertisers, who advertise on the UPC REQUEST™ Central Website(s), would pay an Advertiser Fee based on the time and location that the Web advertisement is displayed.

Usage of the UPC REQUEST™ in Diverse Retail Markets

[0741] The UPC REQUEST™ System can be used to provide consumers quick access to useful product-related information in diverse types of retail shopping environments including, for example, retail superstores, discount department stores, home-improvement stores, computer superstores, drugstores and pharmacies, music stores, video rental stores, bookstores, supermarkets, grocery stores and the like. Each of these retail markets provides a unique environment in which the UPC REQUEST™ System can be used by manufacturers of consumer products to effectively deliver product-related information to consumers in retail stores, at home, in the office or on the road, before and after consumer purchases.

[0742] As the UPC REQUEST™ System provides manufacturers with an effective way and means of making direct contact with present and future customers, it enables both manufacturers and retailers alike to influence demand in ways that have hitherto have been unavailable. Such features of the UPC REQUEST™ System present enormous growth opportunities in retail supply and demand chain management across diverse markets within our ever expanding global economy.

Modifications of the Illustrative Embodiments of the Invention

[0743] The present invention has been described in great detail with reference to the above illustrative embodiments. It is understood, however, that numerous modifications will readily occur to those with ordinary skill in the art having had the benefit of reading the present disclosure.

[0744] For example, in the illustrative embodiments described hereinabove, separate databases are maintained by each data-synchronized IPD Server for (i) registered products within the system, and (ii) non-registered products within the system. Notably, the reasons for using a dual database design of this sort would be based largely on economics, namely: only those companies who have paid the required maintenance (or registration) fees get their products and linked-URLs "registered" with the system, whereas non-paying companies and organizations do not get their products and linked-URLs registered with the system, regardless of how such product-URL information is ascertained (e.g. by solicitation versus data-mining).

[0745] Thus it is contemplated that in some embodiments of the present invention, each IPD Server will be designed to maintain only a single database for maintaining product-URL information currently available on the Internet. In such embodiments of the present invention, the concept of "non-registered" products will be altogether avoided, since the system implementation and administration may be designed not to require companies to pay maintenance (or registra-

tion) fees in order that their products and linked URLs are registered with the IPI system. Instead, some alternative income producing scheme will be used in such embodiments of the present invention (e.g. user fees, subscription fees, Internet browser-licensing fees, etc.) for system maintenance and administration.

[0746] When practicing the system and method of the present invention, it is preferred that the UPC label (with its human-readable UPC number) assigned to the particular product be attached, embossed or otherwise embodied on an accessible surface thereof. In addition to applying the UPC label to the external packaging of the product, it is preferred that the UPC label also be printed on any and all product instructions and manuals provided with the product. In this way, the UPC number can be easily read by a human being and then used to access a desired type of product information using the system and method of the present invention.

[0747] In order that the system hereof can be used to find information pertaining to large products such as automobiles, motorcycles, skidoos, farm machinery, boats, etc., the present invention also contemplates assigning UPNs (e.g. UPC or EAN numbers) to such products and attaching, embossing or otherwise embodying the same on an accessible surface thereof. Also, the UPN label can be printed on all instruction booklets and/or operating manuals normally provided with the product. In this way, information related to any particular product that is posted anywhere on the Internet and linked to URLs registered with the IPD Servers 11 of the system hereof can be readily found using the uniquely assigned UPC number assigned thereto by the manufacturer at the time of sale. Notably, multi-media information about such products can be most helpful in regard to the operation, repair and servicing of such products.

[0748] The system and method of the present invention has been shown to combine the use of UPNs, trademarks and company names when making a product information request of the system. It is understood, however, that the present invention can be practiced using any one of these items of information, alone or in combination with each other, in order to place a product information request with the system hereof.

[0749] Also, while the system of the illustrative embodiment has been shown used to collect, transport and serve information related to consumer products, it is understood that the system can be used to link the URLs of HTML (and other Internet) documents with consumer services assigned uniform service numbers (USN) which may be based on the UPC or EAN numbering system, or some other suitable system. In such alternative embodiments, the IPI Database would contain information pertaining to uniform service numbers (USN) that have been linked to the URLs of HTML or like documents on the Internet by the manufacturer or its agents, in essentially the same manner as conducted for consumer products. Such USN/URL management operations can be carried out in a similar to that described in connection with UPN/URL management along the retail supply and demand chain.

[0750] In connection with the consumer service information embodiment of the present invention, it is understood that at present, few (if any) services have been assigned a UPC (or EAN) number in the manner that nearly all consumer products have been assigned in the contemporary period. In spite of this fact, however, the present invention contemplates the need and utility of widespread assignment of UPC, EAN or similar numbers by service providers to particular services (as well as the imprinting of UPC, EAN or similar symbols on printed service brochures and advertisements. Notably, assigning uniform service numbers (USNs) to particular services, and labeling printed and graphical brochures and advertisements with such universal numbers, will provide a number of new opportunities hitherto unavailable.

[0751] In particular, service-related information could be easily found (i.e. located and accessed) on Web-sites using the system and method of the present invention, and thereafter the service easily procured through an electronic data transaction. In accordance with the present invention, this can be achieved by uniquely identifying and assigning "particular" services by a Universal Service Code (USC) which has many if not all of the attributes of a conventional UPC. While not necessary, a single digit may be optionally added to the USC in order to demark that services, rather than products, are being identified. An example of such USC labeling would be printing an assigned UPC label (number) on: admission tickets to a theatrical, dramatic or musical performance and/or its playbill; admission tickets to a movie; admission tickets to a concert and/or its concert program; admission tickets to a sporting event and/or its sports program; admission tickets to an art, science or history museum; admission tickets to the zoo or botanical gardens; and the like. The UPC label would be encoded to identify a particular event at which a n entertainment, educational or professional service is provided. The UPC label printed on the tangible medium associated with the promotion of or access to the particular service would then be registered with the IPSI Registrant Database of the system hereof, along with the name of the provider of the service, and a list of URLs that identify the Web locations at which particular kinds of information related to the particular service can be found (in accordance with the categories of FIG. 4A2).

Application of the System and Methods of the Present Invention to Print and Electronic Publishing Industries

[0752] Notably, newspaper and magazine articles, product and service brochures, product discount coupons, certificates, documents and other forms of information-carrying products embodied within a print medium (e.g. paper, plastic, metal, glass, etc.) are properly deemed "consumer products" within the scope and spirit of the present invention, regardless of whether of not such products are assigned a UPC or UPC/EAN label by its manufacturer (e.g. publisher/printer). Thus, the system and methods of the present invention described hereinabove can be used to manage and serve print-media related information on the WWW to consumers (e.g. readers), relating to particular publications (e.g. articles, stories, product advertisements, etc.) embodied in a published print-medium.

[0753] Stated more specifically, the system and methods of the present invention can be used to manage symbolic links created between (i) an UPN uniquely assigned to an information carrying product or object embodied in any form of media (e.g. printed publication existing in physical

space, or other form of information carrying media), and (ii) the URL specifying the location of an information resource (e.g. Web document) on the Internet (e.g. WWW). In such application environments, the UPN can be any unique number assigned to the information carrying product embodied in any physical medium. In general, each UPN can be expressed in numerical or alphanumerical form, and in the case of print-type media, will be encoded within a machine-readable structure, such as a bar code symbol structure, associated with the information carrying product. Preferably, each unique UPN will be assigned to the information carrying product by a source of an authority assuming managerial responsibility for the universal product/service (i.e. object) numbering system being used. Such organizations may be public, quasi-public, or private, depending on the circumstances at hand.

[0754] In general, Web-based information resources located at URLs to be symbolically linked to particular UPNs assigned to information carrying products by a particular publisher can be served from any Internet (http) information server, including e-commerce enabled servers **12**, **12'**, **12A** and **12B**, shown in FIGS. **2-1** and **2-2** and described in great detail hereinabove.

[0755] UPN/URL management subsystem **9**, server **33**, administration computer **32**, and the EDI-enabled client subsystems shown in FIGS. **2-1**, **2-2**, **2A**, and **2C**, and described in great detail hereinabove can be used by publishers (e.g. a particular type of product manufacturer) to symbolically link each UPN assigned to particular information carrying products (e.g. newspaper and magazine articles, product advertisements, etc.) to one or more URLs specifying product-related information on the Internet. As shown in **FIG. 2D**, a plurality of publisher-operated client subsystems (i.e. manufacturer-operated client subsystems) are connected to a local or wide area TCP/IP-based network, for the purpose of enabling different departments within the publishing organization (e.g. advertising, world news, business, technology, sports, finance, education, arts and leisure, etc.) manage different types of UPN/URL links based on the type of information contained within the URL-specified information resource on the WWW. In the back-offices of publishing firms (e.g. newspaper publishing houses, magazine publishers, product advertisement flyer publishers/printers, product coupon publishing/marketing firms and the like) UPN/URL data link management operations will be carried out prior to print publication using distributed UPN/URL link management methods.

[0756] In the case of a newspaper or magazine publisher firm, each client computer subsystem within a particular department (e.g. sports, international news, national news, local news, business, arts & entertainment, science & technology, etc.) as shown in **FIG. 2D**, would be provided with one or more of the following software programs, namely: (i) computer-based publishing software having integrated UPN/URL data link management capabilities; and (ii) UPN/URL management software in the form of a stand-alone application or utility.

[0757] Operation of computer-based publishing software with integrated UPN/URL data link management capabilities is shown in FIGS. **2E1** and **2E2**. In general, this publishing software program has at least three modes of operation, namely: (1) a composition/editorial mode illustrated in **FIG. 2E1**, in which literary and graphical content can be composed and edited in much the same way as conventional word processing and graphics programs; (2) a UPN/URL data linking mode illustrated in **FIG. 2E2**, in which the author/composer can create symbolic links between URL indexed sections of Web documents and UPN indexed sections of print-media documents, using drag, drop and click procedures similar to those used in conventional "flow charting" and graphics software programs; and (3) a UPN/URL data link table generation mode, in which a "UPN/URL data link table" is generated for each corresponding set of Web and print-media documents, as shown in **FIG. 2E3**, and is electronically transportable to the UPN/URL database management subsystem **9** hereof using electronic data interchange techniques described in detail hereinabove.

[0758] In the first and second modes of operation, in a split-screen structure **300** as shown in **FIG. 2E1** is displayed, wherein the first (e.g. left) side document frame **301** contains information relating to a particular Web document and has an independent scroll bar **302** within a **304B** windows-type GUI-based data structure, and wherein second (e.g. right) side document frame **303** contains information relating to a particular print-media document and has an independent scroll bar **304** within a windows-type GUI-based data structure. By providing such a split-screen display screen GUIs in both of these modes of operation, the publishing software program of the present invention enables the author to independently scroll literary and graphical context in the Web document frame **301** of the GUI, and the print-media document frame **303** thereof, thereby facilitating alignment of corresponding sections during both the composition/editorial mode of operation as well as during the UPN/URL data linking mode of operation.

[0759] When in the composition/editorial mode, the publishing software program of the present invention enables the electronic layout of a Web-based document or publication (e.g. expressible in HTML or SGML code) **300** in frame **301**, and a print-media based document or publication (e.g. expressible in a desired font) in frame **303**, as shown in **FIG. 2E1**. As shown therein, the Web-based document in frame **301** has different content and advertising sections **306** and **308** associated with each Web-page thereof and each such Web-page being located on the WWW at a particular URL. The print-media based document in frame **303** has different content sections and advertising sections **310** and **312** associated with each printed-page thereof. As illustrated in FIGS. **2E1**, each content section in the Web-based document is assigned a Universal Product Number (UPN) (i.e. UPN-encoded bar code symbol) which is symbolically linked or related to a particular content section on the corresponding Web-page located at a specified URL. Likewise, each advertising section in the Web-based document is assigned a Universal Product Number (UPN) which is symbolically linked or related to a particular advertising section on the corresponding Web-page located at a specified URL. During the composition/editorial mode of operation, content in the Web and print-media documents is displayable in the split-screen display structure **300**, so that a particular content section in a Web document is displayed along side of the corresponding section in a print-media document.

[0760] Typically, the author enters the a URL/UPN (or UPN/URL) data linking mode of operation of the program, shown in **FIG. 2E2**, after authoring or otherwise composing literary and/or graphic context within (i) a specific content section on a particular Web document and (ii) a corresponding content section on a particular print-media document. While in this mode of operation, the author is able to create "UPN/URL data links" between pairs of corresponding content sections, pairs of corresponding advertising sections, and/or pairs of content and advertising sections. In accordance with the principles of the present invention, such UPN/URL data links are created by: (1) drawing graphical boundaries around the content (or advertising) section on a particular Web document using a mouse-pointing device and automatically assigning thereto a relative URL index **400** as shown in FIGS. **2E2** and **2E3**; (2) drawing graphical boundaries around the corresponding content (or advertising) section on the corresponding print-media document and automatically assigning thereto a UPN (i.e. UPN-encoded bar code symbol) as shown in **FIG. 2E2** (e.g. obtained from the system administrator using on-line or real-time UPN assignment methods implemented over the Internet); and (3) then drawing a graphical link between such graphically bounded sections between the Web and print-media documents displayed in frames **301** and **303**. During this mode of operation, the UPN/URL data links are graphically represented as double-arrow type links for the author to review, and are editable in much the same manner that such graphical elements are created and edited. However, when returning to the composition/editorial mode, such graphical links are suppressed, and instead, alpha-numeric type UPN/URL links are displayed on the Web and print-media documents.

[0761] After the Web and print-media documents have been finalized and approved for publishing, the computer-based publishing program of the present invention enters the UPN/URL data link table generation mode, shown in **FIG. 2E3**. In this mode, a UPN/URL data link table **400** is generated for each corresponding set of Web and print-media document, as shown in **FIG. 2E3**. Thereafter, this data table is electronically transportable to the UPN/URL database management subsystem **9** hereof using electronic data interchange techniques described in detail hereinabove. This ensures that the UPN/URL data links are stored in the UPN/URL database management subsystem **9** hereof so that when the UPNs (on published print-media) are entered into the browser of a client subsystem **13** hereof, pointing to a IPI Website of the present invention, the corresponding URL or set thereof is automatically displayed thereon.

[0762] The above-described publishing software program will have great value in publishing applications where both Web and print-media versions of documents, articles and the like are simultaneously published on the WWW and in the world of print-media, as in the case of many newspapers, magazines, journals, subscriptions, product discount flyers, and the like. In such instances, each publication will have numerous "Sections" or "Parts", classified by the type of subject matter being addressed, and contributed to by authors and composers from categorically different departments (e.g. business, sports, world news, local news, arts and entertainment, technology, etc.). Thus, in such instances, a "client" copy of the publishing program of the present invention discussed above will run on different client computing subsystems on the LAN or WAN of the publisher, as shown in **FIG. 2D**, while a "server" copy runs on at least one client computing subsystem on the LAN or WAN. Such a client-server solution enables an editorial manager to oversee the entire Web/print-media publication process, while individual writers are permitted to labor on their literary and graphical art assignments, whatever they may be. In short, the novel publishing tools and methods of the present invention have now made linking the worlds of electronic and print-media simple, efficient and reliable.

[0763] In some publishing applications such as coupons, books, financial instruments, notes, product specifications, and the like, there is a need for an alternative way of and means for creating UPN/URL data links between the print-media and electronic media worlds, and automatically generate UPN/URL data link tables for electronic transport to the UPN/URL database management subsystem **9**. In this alternative embodiment of the present invention, shown in FIGS. **2E4** and **2E5**, the way of and means for UPN/URL data linking is realized by an operating system (OS) based function that enables the author to generate "UPN/URL data links between (i) Web documents authored using a first arbitrary content-creating browsing application (e.g. Netscape Navigator browser, HTML Editor program) running on a computing platform, and (ii) UPN-encoded elements of print-media authored using a second arbitrary content-creating/browsing application (e.g. Adobe® Illustrator drawing program or Microsoft Word® word processing program) running on the same platform.

[0764] In practice, such the UPN/URL data linking capabilities are integrated into the multi-tasking and/or multi-threading operating system (OS) (e.g. MAC OS, Window 2000, MS NT, Linux, etc.) on the client computer used to run the first and second arbitrary content-creating/browsing applications **330** and **332**, shown in **FIG. 2E4**. For illustrative purposes, the firs application **330** is the Netscape Navigator browser, whereas the second application **332** is the Adobe® Illustrator graphics program. UPN/URL data links produced from this OS-based UPN/URL data linking functionality are stored in a UPN/URL data link GUI table **400** shown in **FIG. 2E4**, and are electronically transportable to the UPN/URL database management subsystem **9** using electronic data interchange techniques described hereinabove. The end result of this OS-based UPN/URL data linking function of the present invention is to enable consumers (e.g readers) to create UPN/URL data links between print-media documents a nd corresponding Web-based media documents, and also, to transport the same to the UPN/URL database management subsystem **9** hereof.

[0765] In **FIG. 2E4**, the OS-based UPN/URL data linking function (i.e. utility) is shown in its composition/editorial mode, wherein the GUI-based window for the first content creating/browsing application **330** is displayed on the OS GUI, along with the GUI-based window for the second content creating/browsing application **332**, and the GUI-based window for the UPN/URL Data Link Table/Palette **400**.

[0766] In **FIG. 2E5**, the OS-based UPN/URL data linking function (i.e. utility) is shown in its UPN/URL Data Linking Mode, wherein graphically expressed UPN/URL data links are created in a manner similar to that carried out in the first illustrative embodiment shown in FIGS. **2E1** through **2E3** and described hereinabove.

[0767] In the UPN/URL data linking mode, the UPN/URL table is generated and electronically transported to sub-

system **9** hereof. Preferably, the OS-based UPN/URL data linking utility of the present invention is launchable by selecting a command in a pull-down or like window provided on the OS GUI. Once this utility has been launched, then its different modes of operation can be selected by a designated control button provided on the windows-based GUI-structure for the UPN/URL.

[0768] Notably, by using the EDI-based UPN/URL data link management and serving techniques of the present invention, described in great detail hereinabove, publishers of print-media based products can quickly update the UPN/URL Database Management Subsystem **9** to contain current UPN/URL links created only minutes before hand in the back-offices of the publishing firm (e.g. just after final approval by the chief editor assigned to a particular print-media publication). This enables consumers to automatically link to related URLS at the time distribution of the print publication, using Web-enabled client subsystems **13** in accordance with the principles of the present invention.

[0769] These and all other such modifications and variations are deemed to be within the scope and spirit of the present invention as defined by the accompanying Claims to Invention.

What is claim is:

1. An IP-based network installed in a publishing organization and Internet-based system comprising:

a plurality of publisher-operated client computer subsystems connected to said IP-based network, for enabling different departments within said publishing organization to manage different types of UPN/URL links based on the type of information contained within the URL-specified information resource on the WWW and within the UPN-specified section of an article of print media.

2. The Internet-based system of claim 1, wherein each said publisher-operated computer-based publishing system enables the electronic layout of: (i) a Web-based publication having different content and advertising sections associated with each Web-page thereof and each such Web-page being located on the WWW at a particular URL; and (ii) a print-media based publication having different content sections and advertising sections associated with each printed-page thereof; wherebetween each such content section and advertising section is assigned a Universal Product Number (UPN) which is symbolically linked to a particular content or advertising section on the corresponding Web-page.

3. The Internet-based system of claim 1, wherein each said publisher-operated computer-based publishing system enables the electronic layout of: (i) a Web-based publication having different content and advertising sections associated with each Web-page thereof and each such Web-page being located on the WWW at a particular URL; and (ii) a print-media based publication having different content sections and advertising sections associated with each printed-page thereof; wherebetween each such content section and advertising section is assigned a Universal Product Number (UPN) which is symbolically linked to a particular content or advertising section on the corresponding Web-page.

4. The Internet-based system of claim 3, wherein each said computer-based publishing subsystem is capable of automatically generating UPN/URL data link tables listing the URLs of each Web page symbolically linked to UPN assigned to a corresponding printed media page, and that such UPN/URL data link tables are transportable to a UPN/URL database management subsystem using electronic data interchange techniques, thereby enabling consumers (e.g. readers) to link from print-media to corresponding Web-based media using the UPNs printed on documents.

5. An Internet-based consumer product information system for use in a retail shopping environment, comprising:

an e-mail server; and

a plurality of Web-enabled bar code driven consumer product information kiosks deployed within a retail shopping environment, and each said Web-enabled bar code driven consumer product information kiosk embodying e-mail messaging capabilities which enable consumers to automatically save and link CPI-related Web documents as individual attachments to-a preformatted e-mail envelope that is transmitted from said e-mail server, to a remote e-mail address specified by the consumer within said retail shopping environment.

6. The Internet-based consumer product information system of claim 5, wherein said preformatted e-mail envelope is addressed with the consumer/shopper's home, office or like e-mail address by either reading an e-mail address encoded within a bar code (or magnetic-stripe) structure or manually entering the same within the addressee field, and said e-mail envelope is transported to its destination by manual selection of a "send-button" within the displayed e-mail envelope.

7. The Internet-based consumer product information system of claim 5 which further comprises:

a central e-mail server for collecting copies of e-mail documents or transmitted from the Web/e-mail enabled kiosks within said retail shopping environment.

8. The Internet-based consumer product information system of claim 7, which further comprises a relational data base management subsystem (RDBMS) for use in analyzing compiling and storing data contained within said central e-mail server, and for making such information accessible to retailers and manufacturers alike for use in product marketing, sales forecasting, customer intelligence, and like operations.

9. An Internet-based consumer product information system for use in a retail shopping environment, comprising:

an e-mail server; and

a plurality of Web-enabled bar code driven consumer product information kiosks deployed within a retail shopping environment, and each said Web-enabled bar code driven consumer product information kiosk embodies e-mail messaging capabilities which enable consumers to automatically save and record the URLs of CPI-related Web documents within the message field of a preformatted e-mail envelope that is transmitted from said e-mail server, to a remote e-mail address specified by the consumer within said retail shopping environment.

10. The Internet-based consumer product information system of claim 9, wherein said preformatted e-mail envelope is addressed with the consumer/shopper's home, office or like e-mail address by either reading an e-mail address encoded within a bar code (or magnetic-stripe) structure or manually entering the same within the addressee field, and

said e-mail envelope is transported to its destination by manual selection of a "send" button within the displayed e-mail envelope.

11. The Internet-based consumer product information system of claim 9 which further comprises:

a central e-mail server for collecting copies of e-mail documents (and records thereof) transmitted from the Web/e-mail enabled kiosks within said retail shopping environment.

12. The Internet-based consumer product information system of claim 11, which further comprises a relation data base management subsystem for use in analyzing compiling and storing data contained within said central e-mail server, and for making such information accessible to retailers and manufacturers alike for use in product marketing, sales forecasting, customer intelligence, and like operations which enable more effective marketing of consumer products and services in both physical and electronic forms of commerce.

12. An Internet-based information system, comprising:

an IP-based network installed in a retail shopping environment and connected to the infrastructure of the Internet by way of an IP-based router; and

a plurality of publisher-operated client subsystems connected to said IP-based network, for the purpose of enabling different departments within the publishing organization (e.g. advertising, world news, business, technology, sports, finance, education, arts and leisure, etc.) to manage different types of UPN/URL links based on the type of information contained within the URL-specified information resource on the WWW.

13. The Internet-based information system of claim 12, wherein each said publisher-operated client subsystem has a publishing software program having a composition/editing mode of operation and a UPN/URL data linking mode of operation, wherein said computer-based publishing software program simultaneously enables the composition and edition a Web-based document and corresponding a print-media based document.

14. The Internet-based information system of claim 13, wherein said publishing software program has a split-screen GUI having left-side and right-side display frames, wherein the layout of said Web-based document is displayed in said left-side display frame of said split-screen GUI, and said print-media based document is displayed on said right-side display frame of said split-screen GUI.

15. The Internet-based information system of claim 13, wherein said publishing software program generates, during said UPN/URL data linking mode, a UPN/URL data link table indicating the symbolic links established between particular content sections within said Web-based document and particular content sections within said print-media based document.

16. The Internet-based information system of claim 15, wherein said publishing software program supports electronic data interchange functionalities for transporting said UPN/URL data link table to a UPN/URL database management subsystem.

17. The Internet-based information system of claim 14, wherein said left-side display frame has a first set of independent scroll bars, and said right-side display frame has a second set of independent scroll bars, thereby facilitating alignment of corresponding sections during said composition/editorial mode of operation and said UPN/URL data linking mode of operation.

18. The Internet-based information system of claim 14, wherein during said UPN/URL data linking mode of operation, each content section in said Web-based document is assigned a Universal Product Number (UPN) which is symbolically linked to a particular content section in said corresponding Web-page located at a specified URL.

19. The Internet-based information system of claim 14, wherein said Web-based document also includes one or more advertising sections and said corresponding print-based document also includes one or more advertising sections, and wherein at least one of said content sections in said print-based document is assigned a Universal Product Number (UPN) which is symbolically linked to a particular content section on said corresponding Web-based document.

20. The Internet-based information system of claim 19, wherein at least one of said advertising sections in said print-based document is assigned a UPN which is symbolically linked to a URL in at least one Web-based document published on the Internet.

21. The Internet-based information system of claim 12 which further comprises a network information server on said IP-network for maintaining a secure firewall.

22. An Internet-based client subsystem for operation by a publisher of print-media based documents and Web-based documents, comprising:

a computing platform; and

a publishing software program for executing on said computing platform, and having a composition/editing mode of operation and a UPN/URL data linking, mode of operation, wherein said publishing software program simultaneously enables the composition and edition a Web-based document and corresponding a print-media based document.

23. The Internet-based client subsystem of claim 22, wherein said publishing software program has a split-screen GUI having left-side and right-side display frames, wherein the layout of said Web-based document is displayed in said left-side display frame of said split-screen GUI, and said print-media based document is displayed on said right-side display frame of said split-screen GUI.

24. The Internet-based client subsystem of claim 23, wherein said publishing software program generates, during said UPN/URL data linking mode, a UPN/URL data link table indicating the symbolic links established between particular content sections within said Web-based document and particular content sections within said print-media based document.

25. The Internet-based client subsystem of claim 24, wherein said publishing software program supports electronic data interchange functionalities for transporting said UPN/URL data link table to a UPN/URL database management subsystem.

26. The Internet-based client subsystem of claim 23, wherein said left-side display frame has a first set of independent scroll bars, and said right-side display frame has a second set of independent scroll bars, thereby facilitating alignment of corresponding sections during said composition/editorial mode of operation and said UPN/URL data linking mode of operation.

27. The Internet-based client subsystem of claim 23, wherein during said UPN/URL data linking mode of opera-

tion, each content section in said Web-based document is assigned a Universal Product Number (UPN) which is symbolically linked to a particular content section in said corresponding Web-page located at a specified URL.

28. The Internet-based client subsystem of claim 23, wherein said Web-based document also includes one or more advertising sections and said corresponding print-based document also includes one or more advertising sections, and wherein at least one of said content sections in said print-based document is assigned a Universal Product Number (UPN) which is symbolically linked to a particular content section on said corresponding Web-based document.

29. The Internet-based client subsystem of claim 28, wherein at least one of said advertising sections in said print-based document is assigned a UPN which is symbolically linked to a URL in at least one Web-based document published on the Internet.

30. The Internet-based information system of claim 22, wherein each said publisher-operated client subsystem has a operating system (OS) based UPN/URL data linking functionality having a composition/editing mode of operation and a UPN/URL data linking mode of operation, wherein said OS-based based UPN/URL data linking functionality simultaneously enabling the composition and edition a Web-based document and corresponding a print-media based document.

31. The Internet-based information system of claim 30, wherein said OS-based based UPN/URL data linking functionality enables the simultaneous operation of at least first and second content creating/browsing application on a common computing platform having a GUI, wherein said OS-based based UPN/URL data linking functionality having a composition/editing mode of operation and a UPN/URL data linking mode of operation, wherein said first content creating/browsing application enables the creation and browsing of a Web-based document during said composition/editing mode of operation, whereas said second content creating/browsing application enables the creation and browsing of a print-media based document during said composition/editing mode of operation.

32. The Internet-based information system of claim 30, wherein during said UPN/URL data linking mode of operation, symbolic links can be established between particular content sections within said Web-based document and particular content sections within said print-media based document.

33. The Internet-based information system of claim 30, wherein during said UPN/URL data linking mode of operation, said symbolic links can be drawn graphically while alpha-numeric representations on said symbolic links are created and stored in a UPN/URL data link table.

34. The Internet-based information system of claim 30, wherein said OS-based UPN/URL data linking functionality supports electronic data interchange functionalities for transporting said UPN/URL, data link table to a UPN/URL database management subsystem.

35. The Internet-based information system of claim 34, wherein during said UPN/URL data linking mode of operation, each content section in said Web-based document is assigned a Universal Product Number (UPN) which is symbolically linked to a particular content section in said corresponding Web-page located at a specified URL.

36. The Internet-based information system of claim 34, wherein said Web-based document also includes one or more advertising sections and said corresponding print-based document also includes one or more advertising sections, and wherein at least one of said content sections in said print-based document is assigned a Universal Product Number (UPN) which is symbolically linked to a particular content section on said corresponding Web-based document.

37. The Internet-based information system of claim 36, wherein at least one of said advertising sections in said print-based document is assigned a UPN which is symbolically linked to a URL in at least one Web-based document published on the Internet.

38. A retail-based system for installation in retail store comprising:

a subsystem of shelves for supporting one or more UPN-encoded products; and

a plurality of Web-enabled bar code symbol driven kiosks mounted relative to said subsystem of shelves, for reading UPN-encoded symbols on said UPN-encoded products and automatically linking to a plurality of URLs pointing to one or more Web-based documents on the WWW linked to the read UPN by the manufacturer of the product and/or agents thereof using a manufacturer-operated client computer subsystem connected to the infrastructure of the Internet and having electronic data interchange capabilities.

39. An Internet-enabled bar code driven consumer product information kiosk for deployment within a retail shopping environment, comprising:

a computing platform having embodying e-mail messaging capabilities which enable consumers to automatically save and link CPI-related Web documents as individual attachments to a preformatted e-mail envelope that is transmitted from said e-mail server, to a remote e-mail address specified by the consumer within said retail shopping environment.

40. Internet-enabled bar code driven consumer product information kiosk of claim 39, wherein said preformatted e-mail envelope is addressed with the consumer/shopper's home, office or like e-mail address by either reading an e-mail address encoded within a bar code (or magnetic-stripe) structure or manually entering the same within the addressee field, and said e-mail envelope is transported to its destination by manual selection of a "send" button within the displayed e-mail envelope.

41. Internet-enabled bar code driven consumer product information kiosks for deployment within a retail shopping environment, comprising:

a computing platform having embodying e-mail messaging capabilities which enable embodies e-mail messaging capabilities which enable consumers to automatically save and record the URLs of CPI-related Web documents within the message field of a preformatted e-mail envelope that is transmitted from said e-mail server, to a remote e-mail address specified by the consumer within said retail shopping environment.

42. The Internet-enabled bar code driven consumer product information kiosk of claim **43**, wherein said preformatted e-mail envelope is addressed with the consumer/shopper's home, office or like e-mail address by either reading an e-mail address encoded within a bar code (or magnetic-stripe) structure or manually entering the same within the addressee field, and said e-mail envelope is transported to its destination by manual selection of a "send" button within the displayed e-mail envelope.

* * * * *